PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 17/00 — A1

(11) International Publication Number: WO 99/15989

(43) International Publication Date: 1 April 1999 (01.04.99)

(21) International Application Number: PCT/US98/20159

(22) International Filing Date: 25 September 1998 (25.09.98)

(30) Priority Data:
60/060,655 26 September 1997 (26.09.97) US

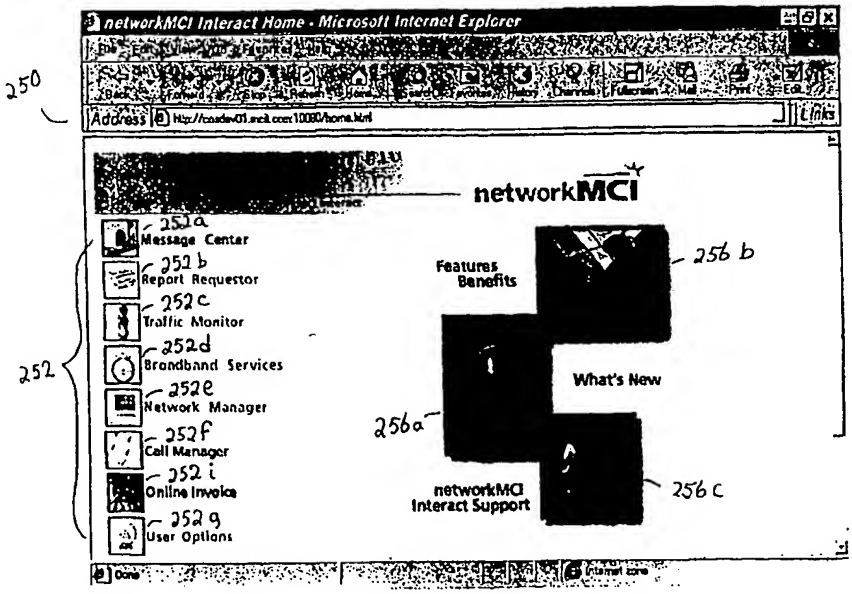
(71)(72) Applicants and Inventors: AHLBERG, Axel, H. [US/US]; 2775 Tartan Lane, Colorado Springs, CO 80920 (US). BECAR, Allyn, P. [US/US]; 6318 Gemfield Drive, Colorado Springs, CO 80918 (US). BRAND, Gregory, L. [US/US]; 6309 Dewsbury Drive, Colorado Springs, CO 80918 (US). FENLEY, Douglas, B. [US/US]; 5205 Zachary Grove #307, Colorado Springs, CO 80919 (US). JONES, Chester, L. [US/US]; 3907 Alemeda Circle, Colorado Springs, CO 80918 (US). WYRICK, Robert, E. [US/US]; 2275 Parliament Drive, Colorado Springs, CO 80920 (US).

(74) Agents: GROLZ, Edward, W. et al.; Scully, Scott, Murphy & Presser, 400 Garden City Plaza, Garden City, NY 11503 (US).

(81) Designated States: AU, BR, CA, JP, MX, SG, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: AUTHENTICATION AND ENTITLEMENT FOR USERS OF WEB BASED DATA MANAGEMENT PROGRAMS



(57) Abstract

An Internet Web-based order entry and system (250) administration system is provided for ordering and fulfilling a suite of Web enabled applications (252). The system includes a capability for enabling customers to order and administer via the Internet. The system is easily accessed and invoked from a generic, off-the-shelf Web browser (250) and at the same time a system infrastructure is provided that enables secure initiation of order entry and system administration to customers from any computer terminal (20) having browser located anywhere in the world.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

AUTHENTICATION AND ENTITLEMENT FOR USERS OF WEB BASED DATA MANAGEMENT PROGRAMS

The present invention relates in general to application software, and more specifically to Web-based system administrative software servicing local and a remote suite of applications in a communications network.

System administrative software is generally known in the information systems industry. This type of software normally provides functions for adding and deleting users, file system management such as backups and version controls, and user security control. The system administrative functions provided by the known software, however, are usually limited to a specific computer platform or even a specific service product. For example, telecommunications service providers offer many different services which have been developed independently over time, and which operate on different computer platforms. Each of the applications associated with different services usually implements its own system administration functions for the related service product separately from the others. Thus, when a customer needs to order or perform administrative functions in more than one service product, the customer is forced to exit the first application before beginning an operation on the second application. Moreover, when using the second service product, the customer must then reenter some of the same information previously entered while using the first service product. Therefore, it is desirable to have a centralized order entry and system administrative infrastructure for handling common functions and associated data for a number of product services.

In addition, with the existing software, a customer support team interaction is usually necessary to complete the order entry and administrative process. Therefore, it is also desirable to have fully automated system administrative software, which automatically connects to the associated back-end systems and updates the back-end databases as necessary. Furthermore, the fully automated order entry system minimizes human intervention during the fulfillment processing for each order entry requested by a customer, resulting in cost and time savings.

In conventional systems, a connection is made with a large legacy system via a dial-up connection from a customer owned personal computer or workstation. This connection frequently, although not always, emulates a terminal addressable by the legacy systems. The dial-up access requires custom software on the customer workstation to provide dial-up services, communication services, emulation and/or translation services and generally some resident

custom form of the legacy application to interface with the midrange or mainframe computer running the legacy system.

There are several problems associated with the approach. First, the aforementioned software is very hardware dependent, requiring multiple versions of software compatible with each of a wide range of workstations customers generally have. Therefore, extensive inventory for distribution becomes necessary. If the customer hardware platform changes through an upgrade, the software licensing issues must be renegotiated. Moreover, installing the software generally requires an intensive effort on the customer and the software support team before any reliable and secure sessions are possible.

Secondly, dial-up, modem, and communications software interact with each other in many ways which are not always predictable to a custom application, requiring extensive trouble shooting and problem solving for an enterprise desiring to make the legacy system available to the customer, particularly where various telephone exchanges, dialing standards or signal standards are involved.

Thirdly, although more businesses are turning to the Internet to improve customer service and lower costs by providing Web-based support systems, when an enterprise desires to make more than one system available to the customer, the custom application for one legacy system is not able to connect to a different legacy system, and the customer must generally logoff and logon to switch from one to the other. The delivery technology used by the two legacy systems may be different, requiring different interface standards, and different machine level languages may be used by the two systems, as for example, the 96 character EBCDIC language used by IBM, and the 127 ASCII character language used by contemporary personal computers. Therefore, an integrated and unified Web-based system for providing an access to a number of different legacy systems in one session is desired.

Finally, the security and entitlement features of the various legacy systems may be completely different, and vary from system to system and platform to platform. It is therefore, desired to provide connectivity to enterprise legacy systems over the public Internet, as the Internet provides access connectivity world wide via the TCP/IP protocol, without need to navigate various telephone exchanges, dialing standards or signal standards.

The popularity of the public Internet provides a measure of platform independence for the customer, as the customer can run their own Internet

Web browser and utilize their own platform connection to the Internet to enable services. This resolves many of the platform hardware and connectivity issues in the customers favor, and leaves the choice of platform and operating system to the customer. Web-based programs can minimize the need for training and support since they utilize existing client software, i.e., a browser, which the user has already installed and already knows how to use. Moreover, there is no longer a need to produce and distribute voluminous hard copies of documentation including software user guides. Further, if the customer later changes that platform, then, as soon as the new platform is Internet enabled, service is restored to the customer. The connectivity and communications software burden is thus resolved in favor of standard and readily available hardware and the browser and software used by the public Internet connection.

An Internet delivered paradigm obviates many of the installation and configuration problems involved with initial setup and configuration of a customer workstation, since the custom application required to interface with the legacy system can be delivered via the pubic Internet and run within a standard Web-browser, reducing application compatibility issues to browser compatibility issues. The Web-based fully automated order entry system simplifies many of the fulfillment issues since the customers need not be provided with additional software and software user guides, other then a web browser.

For the enterprise, the use of off-the-shelf Web browsers by the customer significantly simplifies the enterprise burden by limiting the client development side to screen layout designs and data presentation tools that use a common interface enabled by the Web browser. Software development and support resources are thus available for the delivery of the enterprise legacy services and are not consumed by a need for customer support at the workstation level.

The present invention is to provide a Web-based, on-line application system for processing system administrative and order entry functions for an integrated suite of services and products over the Internet. For example, a suite of products and services may include personal communications services such as pagers, cellular phones, and voice mail, traditional wireline services, and Internet access. Other services include Toll Free Number Network and conferencing. Integrating these products and services and providing an online system for order entry and administration over the Internet allow improved flexibility to a customer when managing their

telecommunications accounts. For instance, the customer may effect desired changes in their accounts, in real time and instantaneously, by utilizing the system of the present invention. Moreover, the customer may instantaneously order additional services over the Internet via the system of the present invention.

The present invention also handles the security and authentication requests from both the client and the server side of the applications implementing the suite of products and services. In this way, a single security profile may be maintained for a given customer, the security profile being included in a centralized database servicing the suite of disparate products and services.

Furthermore, the present invention provides a real time transaction capabilities for notifying the client and server sides of applications implementing the suite of products and services, of the modifications effected by the customer using the system of the present invention. Upon receiving the notifications, the applications may update their data with the modifications effected, such that all data maintained about the customer are kept synchronized at all times.

Additionally, the present invention includes a reconciliation process, which may be run periodically, for example, on a daily basis, for retrieving data from the mainframe database in order to refresh all data relating to customer information, such that the data stored in the present invention is kept up-to-date and synchronized with the mainframe system.

Moreover, the present invention provides a fulfillment process for notifying a fulfillment house of all new customers added to the system. The fulfillment house then may send a welcome package including subscription information to the new customers as necessary. The present invention also includes a billing process for directing billing information to different billing streams as desired by the customer.

The above features provided by the present invention are attained preferably with a Web-based centralized authentication and entitlement administration system for enabling a customer to enter orders over the Internet from a client terminal for one or more application services available at an enterprise Intranet. The system includes a client browser application running in a client workstation and providing an integrated interface to the one or more application services. The client browser application provides a graphical user interface for

interacting with the customer. Additionally, an order entry server located at the enterprise Intranet is provided. The order entry server communicates over the Internet with the client browser application to provide authentication and entitlement information associated with the customer. The client browser application uses the authentication and entitlement information associated with the customer and applies the authentication information in validating the customer before enabling the customer to access the enterprise Intranet. The client browser application enables the customer to access only those application services to which the customer is entitled. An order entry object initiated via the client browser application is also included for enabling presentation of entry options for the customer, the entry options including adding a new order entry, modifying an existing order entry, and canceling an order entry. The order entry object further communicates customer entry of a specific entry option to the order entry server, thus allowing the customer to enter new orders, modify existing orders, and cancel orders for the application services within the customer entitlements via the integrated interface.

Preferred embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Figure 1 illustrates the software architecture component comprising a three-tiered structure;

Figure 2 is a diagrammatic overview of the software architecture of the networkMCI Interact system;

Figure 3 is an illustrative example of a backplane architecture schematic;

Figure 4 illustrates an example client GUI presented to the client/customer as a browser web page;

Figure 5 is a diagram depicting the physical networkMCI Interact system architecture;

Figures 6 and 7 illustrate an architectural overview for the system of the present invention;

Figure 8 illustrates an example of a fulfillment letter sent to a customer;

Figure 9 illustrates message class diagram showing relationship among the classes utilized in the present invention;

Figure 10 is an illustrative example of an order entry process flow in the system of the present invention;

Figure 11 is an input process flow diagram, illustrating inputs to the system of the present invention;

Figure 12 is an output process flow diagram, illustrating outputs from the system of the present invention;

Figure 13 illustrates message formats used in transaction requests and responses communicated to and from the client and server applications in the system of the present invention;

Figure 14 is a diagram illustrating data flow interface with the fulfillment house in the system of the present invention;

Figure 15 is a diagram illustrating data flow interface with the Reporting application;

Figure 16 is a diagram illustrating data flow interface with the Toll Free Network Manager application;

Figure 17 is a diagram illustrating data flow interface with the Broadband application;

Figure 18 is an example of a StarOE main window which is launched from the "networkMCI Interact" home page;

Figure 19 illustrates a dialog box presented to a user from the StarOE main window for selecting an enterprise;

Figure 20 illustrates an example of a new/modify users screen used for setting up new users or modifying options available to the existing users;

Figure 21 illustrates a typical screen for setting up Toll Free Network Manager (TFNM) security;

Figure 22 is a sample StarOE screen for adding and modifying reporting options which are used by the StarWRS;

Figure 23 is a sample of the StarOE access setup screen; and

Figure 24 illustrates a toll free network manager setup dialogue selected from the main window.

An overview of the Web-enabled integrated system

The present invention is one component of an integrated suite of customer network management and report applications using a Web browser paradigm. Known as the networkMCI Interact system ("nMCI Interact") such an integrated suite of Web-based applications provides an invaluable tool for enabling customers to manage their telecommunication assets, quickly and securely, from anywhere in the world.

The nMCI Interact system architecture is basically organized as a set of common components comprising the following:

1) an object-oriented software architecture detailing the client and server based aspect of nMCI Interact;

2) a network architecture defining the physical network needed to satisfy the security and data volume requirements of the networkMCI System;

3) a data architecture detailing the application, back-end or legacy data sources available for networkMCI Interact; and

4) an infrastructure covering security, order entry, fulfillment, billing, self-monitoring, metrics and support.

Each of these common component areas will be generally discussed hereinbelow.

Figure 1 is a diagrammatic illustration of the software architecture component in which the present invention functions. A first or client tier 10 of software services are resident on a customer workstation 10 and provides customer access to the enterprise system, having one or more downloadable application objects directed to front-end business logic, one or more backplane service objects for managing sessions, one or more presentation services objects for the presentation of customer options and customer requested data in a browser recognizable format and a customer supplied browser for presentation of customer options and data to the customer and for internet communications over the public Internet. Additional applications are directed to front-end services such as the presentation of data in the form of tables and charts, and data processing functions such as sorting and summarizing in a manner such that multiple programs are combined in a unified application suite.

A second or middle tier 16, is provided having secure web servers and back-end services to provide applications that establish user sessions, govern user authentication and their entitlements, and communicate with adaptor programs to simplify the interchange of data across the network.

A third or back-end tier 18 having applications directed to legacy back-end services including database storage and retrieval systems and one or more database servers for accessing system resources from one or more legacy hosts.

Generally, the customer workstation includes client software capable of providing a platform-independent, browser-based, consistent user interface implementing objects programmed to provide a reusable and common GUI abstraction and problem-domain abstractions. More specifically, the client-tier software is created and distributed as a set of Java classes including the applet classes to provide an

industrial strength, object-oriented environment over the Internet. Application-specific classes are designed to support the functionality and server interfaces for each application with the functionality delivered through the system being of two-types: 1) cross-product, for example, inbox and reporting functions, and 2) product specific, for example, toll free network management or call management functions. The system is capable of delivering to customers the functionality appropriate to their product mix.

Figure 2 is a diagrammatic overview of the software architecture of the networkMCI Interact system including: the Customer Browser (a.k.a. the Client) 20; the Demilitarized Zone (DMZ) 17 comprising a Web Servers cluster 24; the MCI Intranet Dispatcher Server 26; and the MCI Intranet Application servers 30, and the data warehouses, legacy systems, etc. 40.

The Customer Browser 20, is browser enabled and includes client applications responsible for presentation and front-end services. Its functions include providing a user interface to various MCI services and supporting communications with MCI's Intranet web server cluster 24. As illustrated in Figure 3, the client tier software is responsible for presentation services to the customer and generally includes a web browser 14 and additional object-oriented programs residing in the client workstation platform 20. The client software is generally organized into a component architecture with each component generally comprising a specific application, providing an area of functionality. The applications generally are integrated using a "backplane" services layer 12 which provides a set of services to the application objects that provide the front-end business logic. The backplane services layer 12 also manages the launching of the application objects. The networkMCI Interact common set of objects provide a set of services to each of the applications. The set of services include: 1) session management; 2) application launch; 3) inter-application communications; 4) window navigation among applications; 5) log management; and 6) version management.

The primary common object services include: graphical user interface (GUI); communications; printing; user identity, authentication, and entitlements; data import and export; logging and statistics; error handling; and messaging services.

Figure 3 is a diagrammatic example of a backplane architecture scheme illustrating the relationship among the common objects. In this example, the backplane services layer 12 is programmed as a Java applet which may be loaded and launched by

the web browser 14. With reference to Figure 3, a typical user session starts with a web browser 14 creating a backplane 12, after a successful logon. The backplane 12, inter alia, presents a user with an interface for networkMCI Interact application management. A typical user display provided by the backplane 12 may show a number of applications the user is entitled to run, each application represented by buttons depicted in Figure 3 as buttons 58a,b,c selectable by the user. As illustrated in Figure 3, upon selection of an application, the backplane 12 launches that specific application, for example, Service Inquiry 54a or Event Monitor 54b, by creating the application object. In processing its functions, each application in turn, may utilize common object services provided by the backplane 12. Figure 3 shows graphical user interface objects 56a,b created and used by a respective application 54a,b for its own presentation purposes.

Figure 4 illustrates an example client GUI presented to the client/customer as a browser web page 250 providing, for example, a suite 252 of network management reporting applications including: MCI Traffic Monitor 252c; Call Manager 252f; a Network Manager 252e and Online Invoice 252i. Access to network functionality is also provided through Report Requester 252b, which provides a variety of detailed reports for the client/customer and a Message Center 252a for providing enhancements and functionality to traditional e-mail communications.

As shown in Figures 3 and 4, the browser resident GUI of the present invention implements a single object, COBackPlane which keeps track of all the client applications, and which has capabilities to start, stop, and provide references to any one of the client applications.

The backplane 12 and the client applications use a browser 14 such as the Microsoft Explorer versions 4.0.1 or higher for an access and distribution mechanism. Although the backplane is initiated with a browser 14, the client applications are generally isolated from the browser in that they typically present their user interfaces in a separate frame, rather than sitting inside a Web page.

The backplane architecture is implemented with several primary classes. These classes include COBackPlane, COApp, COAppImpl, COParm. and COAppFrame classes. COBackPlane 12 is an application backplane which launches the applications 54a, 54b, typically implemented as COApp. COBackPlane 12 is generally implemented as a Java applet and is launched by the Web browser 14. This backplane applet is responsible for launching and closing the COApps.

When the backplane is implemented as an applet, it overrides standard Applet methods init(), start(), stop() and run(). In the init() method, the backplane applet obtains a COUser user context object. The COUser object holds information such as user profile, applications and their entitlements. The user's configuration and application entitlements provided in the COUser context are used to construct the application toolbar and Inbox applications. When an application toolbar icon is clicked, a particular COApp is launched by launchApp() method. The launched application then may use the backplane for inter-application communications, including retrieving Inbox data.

The COBackPlane 12 includes methods for providing a reference to a particular COApp, for interoperation. For example, the COBackPlane class provides a getApp() method which returns references to application objects by name. Once retrieved in this manner, the application object's public interface may be used directly.

As shown in Figure 2, the aforesaid objects will communicate the data by establishing a secure TCP messaging session with one of the DMZ networkMCI Interact Web servers 24 via an Internet secure communications path 22 established, preferably, with a secure sockets SSL version of HTTPS. The DMZ networkMCI Interact Web servers 24 function to decrypt the client message, preferably via the SSL implementation, and unwrap the session key and verify the users session. After establishing that the request has come from a valid user and mapping the request to its associated session, the DMZ Web servers 24 re-encrypt the request using symmetric encryption and forward it over a second socket connection 23 to the dispatch server 26 inside the enterprise Intranet.

A networkMCI Interact session is designated by a logon, successful authentication, followed by use of server resources, and logoff. However, the world-wide web communications protocol uses HTTP, a stateless protocol, each HTTP request and reply is a separate TCP/IP connection, completely independent of all previous or future connections between the same server and client. The nMCI Interact system is implemented with a secure version of HTTP such as S-HTTP or HTTPS, and preferably utilizes the SSL implementation of HTTPS. The preferred embodiment uses SSL which provides a cipher spec message which provides server authentication during a session. The preferred embodiment further associates a given HTTPS request with a logical session which is initiated and tracked by a "cookie jar server" 28 to generate a "cookie" which is a unique server-generated key that

is sent to the client along with each reply to a HTTPS request. The client holds the cookie and returns it to the server as part of each subsequent HTTPS request. As desired, either the Web servers 24, the cookie jar server 28 or the Dispatch Server 26, may maintain the "cookie jar" to map these keys to the associated session. A separate cookie jar server 28, as illustrated in Figure 2 has been found desirable to minimize the load on the dispatch server 26. This form of session management also functions as an authentication of each HTTPS request, adding an additional level of security to the overall process.

As illustrated in Figure 2, after one of the DMZ Web servers 24 decrypts and verifies the user session, it forwards the message through a firewall 25b over a TCP/IP connection 23 to the dispatch server 26 on a new TCP socket while the original socket 22 from the browser is blocking, waiting for a response. The dispatch server 26 unwraps an outer protocol layer of the message from the DMZ services cluster 24, and re-encrypts the message with symmetric encryption and forwards the message to an appropriate application proxy via a third TCP/IP socket 27. While waiting for the proxy response all three of the sockets 22, 23, 27 block on a receive. Specifically, once the message is decrypted, the wrappers are examined to reveal the user and the target middle-tier (Intranet application) service for the request. A first-level validation is performed, making sure that the user is entitled to communicate with the desired service. The user's entitlements in this regard are fetched by the dispatch server 26 from the StarOE server 49, the server component of the present invention, at logon time and cached.

If the requestor is authorized to communicate with the target service, the message is forwarded to the desired service's proxy. Each application proxy is an application specific daemon which resides on a specific Intranet server, shown in Figure 2 as a suite of mid-range servers 30. Each Intranet application server of suite 30 is generally responsible for providing a specific back-end service requested by the client, and, is additionally capable of requesting services from other Intranet application servers by communicating to the specific proxy associated with that other application server. Thus, an application server not only can offer its browser a client to server interface through the proxy, but also may offer all its services from its proxy to other application servers. In effect, the application servers requesting services are acting as clients to the application servers providing the services. Such

mechanism increases the security of the overall system as well as reducing the number of interfaces.

The network architecture of Figure 2 may also include a variety of application specific proxies having associated Intranet application servers including: a StarOE proxy for the StarOE application server 39 for handling authentication order entry/billing; an Inbox proxy for the Inbox application server 31, which functions as a container for completed reports, call detail data and marketing news messages; a Report Manager proxy capable of communicating with a system-specific Report Manager server 32 for generation, management and receipt notification of customized reports; a Report Scheduler proxy for performing the scheduling and requests of the customized reports. The customized reports include, for example: call usage analysis information provided from the StarODS server 33; network traffic analysis/monitor information provided from the Traffic view server 34; virtual data network alarms and performance reports provided by Broadband server 35; trouble tickets for switching, transmission and traffic faults provided by Service Inquiry server 36; and toll free routing information provided by Toll Free Network Manager server 37.

As partially shown in Figure 2, it is understood that each Intranet server of suite 30 communicates with one or several consolidated network databases which include each customer's network management information and data. For example, the Services Inquiry server 36 includes communication with MCI's Customer Service Management legacy platform 40(a). Such network management and customer network data is additionally accessible by authorized MCI management personnel. As shown in Figure 2, other legacy platforms 40(b), 40(c) and 40(d) may also communicate individually with the Intranet servers for servicing specific transactions initiated at the client browser. The illustrated legacy platforms 40(a)-(d) are illustrative only and it is understood other legacy platforms may be interpreted into the network architecture illustrated in Figure 2 through an intermediate midrange server 30.

Each of the individual proxies may be maintained on the dispatch server 26, the related application server, or a separate proxy server situated between the dispatch server 26 and the midrange server 30. The relevant proxy waits for requests from an application client running on the customer's workstation 10 and then services the request, either by handling them internally or forwarding them to its associated Intranet application server 30. The proxies additionally receive

appropriate responses back from an Intranet application server 30. Any data returned from the Intranet application server 30 is translated back to client format, and returned over the internet to the client workstation 10 via the Dispatch Server 26 and at one of the web servers in the DMZ Services cluster 24 and a secure sockets connection. When the resultant response header and trailing application specific data are sent back to the client browser from the proxy, the messages will cascade all the way back to the browser 14 in real time, limited only by the transmission latency speed of the network.

The networkMCI Interact middle tier software includes a communications component offering three (3) types of data transport mechanisms: 1) Synchronous; 2) Asynchronous; and 3) Bulk transfer. Synchronous transaction is used for situations in which data will be returned by the application server 40 quickly. Thus, a single TCP connection will be made and kept open until the full response has been retrieved.

Asynchronous transaction is supported generally for situations in which there may be a long delay in application server 40 response. Specifically, a proxy will accept a request from a customer or client 10 via an SSL connection and then respond to the client 10 with a unique identifier and close the socket connection. The client 10 may then poll repeatedly on a periodic basis until the response is ready. Each poll will occur on a new socket connection to the proxy, and the proxy will either respond with the resultant data or, respond that the request is still in progress. This will reduce the number of resource consuming TCP connections open at any time and permit a user to close their browser or disconnect a modem and return later to check for results.

Bulk transfer is generally intended for large data transfers and are unlimited in size. Bulk transfer permits cancellation during a transfer and allows the programmer to code resumption of a transfer at a later point in time.

Figure 5 is a diagram depicting the physical networkMCI Interact system architecture 10. As shown in Figure 5, the system is divided into three major architectural divisions including: 1) the customer workstation 20 which include those mechanisms enabling customer connection to the Secure web servers 24; 2) a secure network area 17, known as the DeMilitarized Zone "DMZ" set aside on MCI premises double firewalled between the both the public Internet 25 and the MCI Intranet to prevent potentially hostile customer attacks; and, 3) the MCI Intranet Midrange Servers 30

and Legacy Mainframe Systems 40 which comprise the back-end business logic applications.

As illustrated in Figure 5, the present invention includes a double or complex firewall system that creates a "demilitarized zone" (DMZ) between two firewalls 25a, 25b. In the preferred embodiment, one of the firewalls 29 includes port specific filtering routers, which may only connect with a designated port on a dispatch server within the DMZ. The dispatch server connects with an authentication server, and through a proxy firewall to the application servers. This ensures that even if a remote user ID and password are hijacked, the only access granted is to one of the web servers 24 or to intermediate data and privileges authorized for that user. Further, the hijacker may not directly connect to any enterprise server in the enterprise intranet, thus ensuring internal company system security and integrity. Even with a stolen password, the hijacker may not connect to other ports, root directories or applications within the enterprise system.

The DMZ acts as a double firewall for the enterprise intranet because the web servers located in the DMZ never store or compute actual customer sensitive data. The web servers only put the data into a form suitable for display by the customer's web browser. Since the DMZ web servers do not store customer data, there is a much smaller chance of any customer information being jeopardized in case of a security breach.

As previously described, the customer access mechanism is a client workstation 20 employing a Web browser 14 for providing the access to the networkMCI Interact system via the public Internet 15. When a subscriber connects to the networkMCI Interact Web site by entering the appropriate URL, a secure TCP/IP communications link 22 is established to one of several Web servers 24 located inside a first firewall 25a in the DMZ 17. Preferably at least two web servers are provided for redundancy and failover capability. In the preferred embodiment of the invention, the system employs SSL encryption so that communications in both directions between the subscriber and the networkMCI Interact system are secure.

In the preferred embodiment, all DMZ Secure Web servers 24 are preferably DEC 4100 systems having Unix or NT-based operating systems for running services such as HTTPS, FTP, and Telnet over TCP/IP. The web servers may be interconnected by a fast Ethernet LAN running at 100 Mbit/sec or greater,

preferably with the deployment of switches within the Ethernet LANs for improved bandwidth utilization. One such switching unit included as part of the network architecture is a HydraWEB™ unit 45, manufactured by HydraWEB Technologies, Inc., which provides the DMZ with a virtual IP address so that subscriber HTTPS requests received over the Internet will always be received. The HydraWEB™ unit 45 implements a load balancing algorithm enabling intelligent packet routing and providing optimal reliability and performance by guaranteeing accessibility to the "most available" server. It particularly monitors all aspects of web server health from CPU usage, to memory utilization, to available swap space so that Internet/Intranet networks can increase their hit rate and reduce Web server management costs. In this manner, resource utilization is maximized and bandwidth (throughput) is improved. It should be understood that a redundant HydraWEB™ unit may be implemented in a Hot/Standby configuration with heartbeat messaging between the two units (not shown). Moreover, the networkMCI Interact system architecture affords web server scaling, both in vertical and horizontal directions. Additionally, the architecture is such that new secure web servers 24 may be easily added as customer requirements and usage increases. The use of the HydraWEB™ enables better load distribution when needed to match performance requirements.

As shown in Figure 5, the most available Web server 24 receives subscriber HTTPS requests, for example, from the HydraWEB™ 45 over a connection 44a and generates the appropriate encrypted messages for routing the request to the appropriate MCI Intranet midrange web server over connection 44b, router 55 and connection 23. Via the HydraWEB™ unit 45, a TCP/IP connection 38 links the Secure Web server 24 with the MCI Intranet Dispatcher server 26.

Further as shown in the DMZ 17 is a second RTM server 52 having its own connection to the public Internet via a TCP/IP connection 48. This RTM server provides real-time session management for subscribers of the networkMCI Interact Real Time Monitoring system. An additional TCP/IP connection 48 links the RTM Web server 52 with the MCI Intranet Dispatcher server 26.

With more particularity, as further shown in Figure 5, the networkMCI Interact physical architecture includes three routers: a first router 49

for routing encrypted messages from the Public Internet 15 to the HydraWEB™ 45 over a socket connection 44; a second router 55 for routing encrypted subscriber messages from a Secure Web server 24 to the Dispatcher server 26 located inside the second firewall 25b; and, a third router 65 for routing encrypted subscriber messages from the RTM Web server 52 to the Dispatcher server 26 inside the second firewall. Although not shown, each of the routers 55, 65 may additionally route signals through a series of other routers before eventually being routed to the nMCI Interact Dispatcher server 26. In operation, each of the Secure servers 24 function to decrypt the client message, preferably via the SSL implementation, and unwrap the session key and verify the users session from the COUser object authenticated at Logon.

After establishing that the request has come from a valid user and mapping the request to its associated session, the Secure Web servers 24 will re-encrypt the request using symmetric RSA encryption and forward it over a second secure socket connection 23 to the dispatch server 26 inside the enterprise Intranet.

As described herein, the data architecture component of networkMCI Interact reporting system is focused on the presentation of real time (un-priced) call detail data, such as provided by MCI's TrafficView Server 34, and priced call detail data and reports, such as provided by MCI's StarODS Server 33 in a variety of user selected formats.

All reporting is provided through a Report Requestor GUI application interface which support spreadsheet, a variety of graph and chart type, or both simultaneously. For example, the spreadsheet presentation allows for sorting by any arbitrary set of columns. The report viewer may also be launched from the inbox when a report is selected.

A common database may be maintained to hold the common configuration data which may be used by the GUI applications and by the mid-range servers. Such common data includes but are not limited to: customer security profiles, billing hierarchies for each customer, general reference data (states, NPA's, Country codes), and customer specific pick lists: e.g., ANI's, calling cards, etc.. An MCI Internet StarOE server manages the data base for the common configuration of data.

Report management related data is also generated which includes 1) report profiles defining the types of reports that are available, fields for

the reports, default sort options and customizations allowed; and 2) report requests defining customer specific report requests including report type, report name, scheduling criteria, and subtotal fields. This type of data is generally resident in a Report Manager server database and managed by the report manager.

The Infrastructure component of the nMCI Reporting system includes mechanisms for providing secure communications regardless of the data content being communicated. The nMCI Interact system security infrastructure includes: 1) authentication, including the use of passwords and digital certificates; 2) public key encryption, such as employed by a secure sockets layer (SSL) encryption protocol; 3) firewalls, such as described above with reference to the network architecture component; and 4) non-repudiation techniques to guarantee that a message originating from a source is the actual identified sender. One technique employed to combat repudiation includes use of an audit trail with electronically signed one-way message digests included with each transaction.

Another component of the nMCI Interact infrastructure includes order entry, which is supported by the present invention, the Order Entry ("StarOE") service. The general categories of features to be ordered include: 1) Reporting, including inbound traffic standard, inbound traffic call detail, inbound traffic exception, inbound traffic real time monitor reports, priced reports, and priced call detail reports; 2) Broadband, including basic, standard, enhanced with SNMP, enhanced with adhoc, premium, dedicated SNMP, and dedicated SNMP with adhoc; 3) Toll Free Network Manager; and 4) Call Manager. The order entry functionality is extended to additionally support 5) Event Monitor; 6) Service Inquiry; 7) Outbound Network Manager; and, 8) Online invoicing.

The self-monitoring infrastructure component for nMCI Interact is the employment of mid-range servers that support SNMP alerts at the hardware level. In addition, all software processes must generate alerts based on process health, connectivity, and availability of resources (e.g., disk usage, CPU utilization, database availability).

The Metrics infrastructure component for nMCI Interact is the employment of mechanisms to monitor throughput and volumes at the Web servers, dispatcher server, application proxies and mid-range servers. Metrics monitoring helps in the determination of hardware and network growth.

To provide the areas of functionality described above, the client tier 10 is organized into a component architecture, with each component

providing one of the areas of functionality. The client-tier software is organized into a "component" architecture supporting such applications as inbox fetch and inbox management, report viewer and report requestor, TFNM, Event Monitor, Broadband, Real-Time Monitor, and system administration application. Further functionality integrated into the software architecture includes applications such as Outbound Network Manager, Call Manager, Service Inquiry and Online invoicing.

StarOE

The present invention is directed to a system administration and order entry component (StarOE) of the above described integrated suite of customer network management and report application, referred to as the "networkMCI Interact". The system of the present invention, the StarOE, is used to order, fulfill, and bill for, as well as administer, the suite of network applications, providing a horizontal service for use by all applications. The applications communicate to StarOE for all authentication, entitlement and system administration as well as order entry services. StarOE centrally processes these service requests for the individual applications by providing all order entry and security information for the "networkMCI Interact" suite of applications.

The security information which the StarOE maintains and provides describes identification, authentication and access control used in the suite of applications. All access to the "networkMCI Interact" is controlled by userids and passwords. In addition, individual users are specifically granted access to only the necessary system objects, i.e., file, programs, menus, reports, etc. Access to these individual objects are based upon the customer privilege models, referred to as entitlements, stored in a StarOE database. Thus, all information regarding customers and their access levels for each product in the suite of network applications to which the customers have subscribed are stored in a customer security profile database local to the StarOE. Accordingly, StarOE provides the ability to prevent unauthorized, non-customer access to "networkMCI Interact" data and applications; ability to allow customers to access multiple enterprises with one userid; ability to restrict authorized users to specific Intranet applications and databases based on applications ordered by the customer; and the ability for users to restrict view and/or update capabilities within an application or data set, i.e., customers may

provide or restrict views of their "enterprise" data to subgroups within their organization.

By utilizing the system of the present invention, customers no longer have to place manual calls to order entry hubs when requesting order transactions. For example, users may be added to the system without an enterprise's support team intervention. In sum, customers may manage their communications services in a secure environment and also, for example, monitor their network traffic via the Internet, as well as have a capability to add products and services to their account, in an automated fashion and all in one session without having to enter and exit the individual application services separately, and without having to contact a customer support representative.

Figures 6 and 7 illustrate an architectural overview of the administration and order entry system of the present invention, i.e., StarOE. The StarOE includes a server 202 typically resident in a midrange computer, and a client application 204 running in a user platform having a Web browser, hereinafter referred to as a StarOE client application. The StarOE processes a number of transaction requests relating to authentication and entitlements, from other application services, both from the client and the server 208 sides of the network. In addition, the StarOE server 202 receives transaction requests from the StarOE client. The transactions are typically message driven and comprise requesting transactions and response transactions. The transaction messages will be described in detail in reference to Figure 13. The StarOE server 202 responds to the requests by formulating transaction responses and transmitting them to the requesting servers and clients.

The StarOE client application

The StarOE client application 204 is typically a Web-based GUI interface running in a World Wide Web browser (Microsoft Internet Explorer 4.0.1 or higher), and implemented accordingly and conforming to a standard defined for the GUI interface for the integrated suite of customer network management and report applications ("networkMCI Interact"). The client application's user interface is consistent with other products and services under the suite of network applications.

The client browser application 204 generally includes a Web browser and Java applications and applets for providing a common Web-based GUI for interacting with customers at the front-end side. A backplane unit, a backbone for applications running in

the client platform is typically launched as a Java applet from a Web page, i.e., the "networkMCI Interact" home page (Figure 4). The StarOE client application may then be invoked from the home page by the backplane unit at customer's initiation.

When a customer launches the StarOE application from the home page, the main window as illustrated in Figure 18, is presented. From this main window 500, a customer may select to order and fulfill application services, request user id's, and create user security profiles for the "networkMCI Interact" suite of applications. The main window 500 includes a menu bar 506 with options to perform various StarOE tasks. The main window also includes a toolbar 504, common to all "networkMCI Interact" applications. The toolbar 504 has buttons that also perform the various StarOE functions. Typically, the user list is presented, i.e., displayed as a tree 502, within the main window 500.

The menu options 506 include: file menu options which includes a select enterprise option for allowing administrators to open a user list for a different enterprise, or add a new enterprise to their enterprise list, print option, and exit option which shuts down the StarOE application; edit menu option which includes add new application, modify, and delete options; options menu which enables a global security setup for the toll free manager application; view menu which includes options to refresh the screen by retrieving the latest user list for the opened enterprise from the StarOE server and displaying the list on the screen, to expand all nodes in the user list, and to collapse all nodes in the user list; and help menu option which launches the help engine with StarOE help text. The toolbar 504 also includes the options for a select enterprise, refresh, expand all, collapse all, print and help options.

A typical process flow logic for StarOE client application starts with the home page launching the StarOE client and passing a reference to a common user information object. This object includes the user id, and the default enterprise for that user. The main window 500 having the menu options 506 and the toolbar 504 is then presented. The StarOE client application then sends a "get StarOE security" message by including user id, enterprise id, and the StarOE application code in the message. The StarOE server 202 returns racf id, an access level representing whether the user is an external admin, a member of an account team, an internal admin, or a customer support admin, for example. If the user that launches the StarOE application is an external admin, the user list is displayed immediately since external administrators

may view only one enterprise. For external administrators, an enterprise name is retrieved from the StarOE server 202 by sending and receiving a "get user enterprise list" transaction request and response.

If the user is not an external administrator, then a dialog is presented for the user to select which enterprise to view. When user selects an enterprise to view, a "get user list" message having enterprise id is sent to the StarOE server 202 to retrieve a list of user ids, a list of applications for each user, an access type for each application, and reporting types for StarWRS (e.g., Toll Free, Vnet, Vision, CVNS). The client application also sends a "get application list" message to retrieve from the StarOE server 202 a list of application codes, description, and an application array position. The user list is then displayed within the main window as shown at 502.

Every user list has a New User node 502a as the first node under an enterprise 502b. This node may be selected to order a new user. An existing user node 502c may be selected to edit and add new applications for that user. When an existing user node 502c is selected, the edit/add new application options on the menu 506 is enabled and disabled according to what applications the user already has. An existing user application node 502d may be selected to edit/modify/delete options within the application.

Figure 19 illustrates a dialog box 510 presented to a user for selecting an enterprise. This dialog box is displayed when the user chooses the select enterprise menu option or toolbar button. It may also be displayed automatically when the StarOE client application determines that the user is not an external user who, therefore, may have access to multiple enterprises ids. The purpose of the dialog is to allow the administrator to work with a different enterprise, as well as add an enterprise to their enterprise list.

Typically, the dialog is presented with a list of enterprises to which that the user has access as shown at 510. An object constructor takes a user id as an argument and formats and sends a "get user enterprise list" message to the StarOE server 202. The StarOE server 202 returns the list of enterprises for the user to the StarOE client application 204 which then displays the list as shown at 510. The select button allows the administrator to add an enterprise by causing the client application 204 to send an "add user enterprise" message to the StarOE server 202. The search button 514b allows the administrator to search for an enterprise by entering

either an enterprise id or part of the enterprise name with wild card characters. A selection of the search button 514b also causes the client application 204 to send a "get enterprise list" to the StarOE server 202. The delete button 514c when pressed, removes the enterprise from the administrator's enterprise list. The cancel button 514d is used to return control to the main window 500.

Figure 20 illustrates an example of a new/modify users screen 520 used for setting up new users or modifying various options available to the existing users. The screen is typically presented as a modal dialog displayed on top of the main window when a user node 502c is selected and the modify option is chosen from the main window 500. It may also be presented when the new user node 502a and the add new application menu option is chosen. This screen 520 is generally presented by the client application 204 invoking a class object for handling the new/modify user functionality. On construction of this object, an add/modify flag is initialized. If an add flag is set, an empty screen is displayed with an empty user profile in memory and a new password is generated and displayed in the password field as shown at 522. If a modify flag is set, the client application 204 via the object it invoked, formats and sends a "get user info" message to the StarOE server 202. When a response to the message is received, the client application 204 stores user profile received with the response from the StarOE server 202 in memory, and populates the fields on the screen 520 accordingly. The application setup button 524a may be enabled during an add user process for presenting the appropriate application's security setup screen. The cancel button 524c generally prompts the user for confirmation to cancel, and returns control to the main window 500. When a submit button 524b is pressed, the user profile information is sent to the StarOE server 202 using the "set user info" message. This message is always sent during an add process. During a modify process, this message is sent to the StarOE server 202 only if any changes were entered by the user. When the StarOE server 202 receives the message, it returns an acknowledgment message. The client application 204 then returns control to the main window 500.

During the StarOE add or modify procedure described above, security information regarding customer entitlements for application services may also be initialized. Figure 21 illustrates a typical screen for setting up Toll Free Network Manager (TFNM) security information. This screen 530 is typically displayed when TFNM is ordered or modified. The corp

ids may be displayed in a tree format as shown at 532. If a corp id is part of the user's profile, a racf id may be displayed with the corp id in the tree. The user id is typically displayed in the title bar. A user's TFNM security profile includes at least one corp id, with each corp id having an associated racf id. The racf ids 534 are manually entered for each corp id. A default corp id is chosen for each user.

A setup security object typically handles the process of setting up security for each application. A constructor for this object initializes the user id and a modify flag as passed in from the StarOE client application 204. The object retrieves the toll free hierarchy from the StarOE server 202 using the "get hierarchy" message. The client application 204 sends the enterprise id, and tollfree flag in the request, and the StarOE server 202 returns the list of tollfree corp ids for the enterprise. If the modify flag is set, a "get security" message is sent to the server 202 to retrieve the user's TFNM security profile. As the tree is loaded with each toll free corp id 532, racf id 534 is entered by a user. When the submit button is pressed, the setup security object calls its send security method which causes the formatting and sending of "setTFNM security" message to the StarOE server 202. When the StarOE server 202 receives the message, it sets the security accordingly for the TFNM application.

The StarOE system is also utilized to order, for example, to add or modify, various reporting options used during report generation by the StarWRS. The StarWRS application is a reporting mechanism for the "networkMCI Interact". Figure 22 is a sample StarOE screen 540 for adding and modifying reporting options which are used by the StarWRS. The StarOE displays the toll free hierarchy for security setup when toll free reporting is ordered or modified. The hierarchy includes a list of corp ids for a given enterprise, with each corp id 542 having a list of toll free numbers 544 under it. The list may be displayed in a tree format, similar to the TFNM security display 530. The reporting options at the toll free number level include unpriced reports, unpriced call detail, and real time monitor reports.

Typically, a user's toll free reporting security profile includes at least one toll free number with at least one reporting option associated. The client application 204 generally invokes an object to handle the reporting option changes and passes in the user id and a modify flag. This object then retrieves the toll free hierarchy from the StarOE server 202 using the "get hierarchy" message. The

client application 204 sends the enterprise id, and toll free flag in the request, whereby, the server 202 returns the list of toll free corp ids for the enterprise. If a modify flag is set, a "get tollfree security" message is sent to the server to retrieve the user's toll free security profile.

As each corp id is expanded, a "get tollfree numbers" message is sent to the server 202 asking for all the numbers for the corp id selected. As each toll free number is added, a search in the user's profile for that number is conducted. If the number is found, the report options are added next to number text as shown at 546. Furthermore, if the number has been deactivated, a text "<inactive>", for example, is added to the display as shown at 548. The inactive numbers are not modifiable. When the unpriced reports or unpriced call detail check boxes are changed, the text next to the toll free numbers selected reflects the state of the check box. The check boxes depict report options to which a user has access for toll free numbers. When more than one toll free number is selected, the check boxes are marked unchecked. When a submit button is pressed, the object calls its send security class method which causes the formatting and sending of a "set user tollfree security" message to the StarOE server 202.

In addition to handling the orders for the suite of applications in the "networkMCI Interact", the order entry application, StarOE, may also be modified via the StarOE. Figure 23 is a sample of the StarOE access setup screen 550. The StarOE order object is created any time the StarOE node for an existing user is selected and the edit/modify menu option is selected from the main window 500. It may also be presented when the Application Setup button 524a is pressed on the new user screen 520.

Typically, the StarOE client application 204 invokes a class object for handling the StarOE order functionality. The object's constructor initializes the user id of the user being configured, a modify flag, and the StarOE level of the logged in administrator as passed in from the client application 204. User level options 552 may be set according to the administrator's StarOE access level. The administrator may not select an access level that is higher than the present level. For example, an account team may not change a user's access to an internal level. When a submit button 554 is pressed, the racf id is checked for a valid value, then the object calls the iSendSecurity() method which formats and sends a "set StarOE Security" message to the StarOE server 202.

The Racf id represents the user's mainframe id. The user level is used to determine the level of access the user has within StarOE. An administrator may have one of the following access levels: external, account, and internal. External administrators have access to view and update all users under their enterprise. They may be able to modify a user's security for the applications that they have permission for administration. These users may not be allowed to order applications, but may need to request their orders through their account team representative. An account administrator is used for account team members. This level of user may have the ability to view user lists for multiple enterprises, and configure a user's application access information. Internal users have the same privileges as account administrators, as well as the ability to order new applications for new and existing users. Internal administrators also have the ability to setup global security information for each application. The internal administrators have access to multiple enterprise ids.

The StarOE is also utilized to setup global security for corp ids. The purpose for setting up global security is to allow the administrator to make corp ids available for inclusion in users' profiles. Figure 24 illustrates a toll free network manager setup dialogue selected from the main window 500. Initially, the toll free corp id list is retrieved from the server 202 via a "get hierarchy list" message. The toll free numbers are placed in appropriate boxes 562a,b,c, based on the flag returned with the hierarchy list, i.e., ' ' for corp ids, '8' for TFNM, and 'E' for enterprise. Corp ids in the corp id box 562a are those toll free corp ids for a given enterprise which have not been made TFNM participants and TFNM enterprise participants. The TFNM list 562b includes all toll free corp ids for a given enterprise that have been made TFNM participants. The enterprise list 562c includes all toll free corp ids for a given enterprise that have been made an enterprise level TFNM participant. Once the corp ids in the list are placed in the appropriate boxes, a user may then move the corp ids among the list boxes. For example, if a corp id is in the "corp id" list 562a, it may be moved to the "TFNM" list box 562b. Corp ids in the "TFNM" list box 562b may be moved to the "enterprise" list box 562c. Pressing the submit button 564 causes a "set TFNM corpid" message to be sent to the StarOE server 202. The server 202 then notifies and sends any corp ids moved among the list boxes to the mainframe registry. The corp ids are then added to the user's profile. Figures 21, 22,

and 24 illustrated screen displays for performing various order entry and administrative functions for the TFNM. Other application services, including reporting for VNET, Vision, Broadband, Call Manager, and invoice reporting may be ordered and the security information pertaining to each application may be modified in a similar manner.

The screens illustrated in Figures 18-24 and associated class objects are invoked from the StarOE client application 204, which is launched by the backplane unit as described above. The StarOE client application 204 employs a Java application program and is implemented by extending the COApp class, one of common objects provided and utilized in the present invention. Because the client program 204 is not implemented as an applet, and also because the client program 204 employs the container Frame for customer display windowing purposes, the client program 204 runs, to a degree, independent of the browser within which the backplane is deployed.

The StarOE client application interacts with the StarOE server in providing various order entry functions for all applications as described above and will be described below in reference to the back-end functionality of the StarOE. Communications between the StarOE client 204 and the server 202 typically use TCP/IP, running a UNIX process listening in on a known TCP port.

The StarOE server

The StarOE server 202 includes a number of processes for performing a number of specific functions. For example, a fulfillment process monitors new customers being added to the system and notifies a fulfillment house accordingly. The fulfillment house then may send appropriate subscription packages according to the information received from the fulfillment process to the new customer. Another process, a reconciliation process, may handle synchronization of data with a mainframe system database and also with databases associated with the individual fulfilling systems. Yet another process, a billing process, may handle directing billing information to different billing streams.

The OE server 202 stores all the "networkMCI Interact" users and their security information such as passwords, application entitlements and hierarchies which may be requested by other application servers and clients in the network. Each user or customer is given a username and a password for accessing the "networkMCI Interact". In addition, during the order entry process, each user in the "networkMCI Interact" is provided with entitlement and hierarchies

describing the user's access privileges to specific application services and also to various sub-services available within the applications.

Generally, the hierarchies are customer-defined during the order entry process, and describe the subdivision of calls into nodes arranged in a n-way tree. The "networkMCI Interact" back-end servers apply the hierarchy definitions to their data at report time when generating reports, typically as queries on a node-by-node basis to the result data set which was extracted using any other criteria supplied. The trees of the hierarchies have essentially arbitrary complexity, i.e., the number of nodes is unlimited. Each node is assigned calls according to a template of conditions. Conditions may be defined as a combination of one or more ANIs, corp IDs, ID codes, Card numbers, account codes, location/node ids, etc. These filters may be applied at any node in the tree. The hierarchies may be applied as both selection criteria (e.g., "report on all calls at these nodes or their descendants", in combination with other criteria) and roll-up targets (e.g., group the results in this report at this level in the tree). These entitlement and hierarchies may be modified via the StarOE client application executed in the customer workstation 204.

An example of a service provided by the StarOE is an authentication process during a "networkMCI Interact" customer login procedure. As described above with reference to the StarOE client application, an initial access to the suite of network applications ("networkMCI Interact") is obtained through the Web pages invoking a login process and a backplane logic. The login process typically begins with a customer entering a username and password pair on an initial Web page retrieved from the Web site providing the suite of network applications. Referring to Figures 6 and 7, a process running in a client platform sends transaction request messages via the Dispatcher 206 to the OE server 202. The StarOE server 202 generally responds to requests by searching the security profile for the information requested, formulating appropriate transaction response messages and transmitting them back to the requesting process. More particularly, during the login procedure, the client login process formulates a transaction message including a user name/password and a validation request for a given customer. The StarOE server 202 looks for the matching name/password pair in the security profile for the customer, and if the name/password pair is found, the server 202 formulates a valid user message response for the login process running in the client platform, including in the

message the enterprise id, time zone, and user id information, and transmits the response via TCP/IP back to the login process. In addition, each user password maintained in StarOE is set to expire at a predefined interval, e.g., every 90 days. Accordingly, when the StarOE server 202 detects that the password has expired, the server 202 notifies the customer, via the client application 204 to change the password. The changed password is sent to the StarOE server 202 formatted in a message interface, "change password request," for example. The server 202 upon receiving the message updates the password for the given user in its user profile typically stored in the database 314, and responds with appropriate return codes to the StarOE client 204. The login process, upon receiving the response may then continue with its normal course of processing.

Another example of a service provided by the StarOE is retrieving an application entitlement list for a given customer. As described briefly above, an entitlement describes a privilege or authorization that a customer has. It is akin to the access levels in UNIX which are granted when a customer belongs to certain user groups. It describes what applications a customer may access. It also describes what the customer may be allowed to do within that application. In addition, it describes what back-end services that application and customer combination may access. For example, a customer may be entitled to use or access many applications and for each application, the customer may have a different set of entitlements. These entitlements may come in two different sets. The first specifies what the customer may do with this application. For example, it may allow the customer to have update access to a particular view and only read-only access in a different view. The second set of entitlements specify what back-end services this particular application and customer may access.

As described previously, all the information relating to entitlements for a given customer is stored in customer profile database 314 located with the StarOE server. When the backplane requests via TCP/IP the entitlement transaction, for example, in a get application list request message, the security module retrieves and transmits back via TCP/IP to the backplane the list of authorized applications accessible by a given customer in a transaction response. The backplane uses the list to determine which buttons on the "networkMCI Interact" home page should be activated, thus controlling access to products. Similarly, individual back-end application servers 208 may make a request for entitlements within that application for a given customer. For example,

the StarWRS, a "networkMCI Interact" reporting system, generates a request for hierarchy data for VNET, VISION, and Toll-free customers whenever reports need be generated. In response, the StarOE retrieves the corresponding hierarchy data and transmits them in real time to StarWRS.

The Outbound Network Manager (ONM), for managing the customer's private networks such as VNET and Vision networks, is another example of an application service having application specific customer entitlements defined and maintained in the StarOE database 314. For instance, the customers ordering ONM sign up at the corp id level. Each user may be associated with multiple corp ids. In addition, each user may have one or more of the entitlement features within ONM, including: CPN Order, Calling Card Order, Dialing Plan Order, ID Code Set Order to which orders a user's access may be permitted or restricted; Approve Order signifying approval authority in creating orders; Modify Order for defining user's authority to modify/delete orders.

In providing the authentication, entitlement, and hierarchy information, including those described above, the StarOE server 202 includes the StarOE database 314 for storing user profiles locally. Additional data needed are typically accessed from the enterprise host systems 320. The StarOE server 202 may be implemented as a concurrent server allowing simultaneous multiple client connections. The server 202 listens on a known port, and when a client connects, the server 202 forks a new process to handle the client connection. The server 202 may also implement a thread to handle each client connection.

The StarOE server 202 may be implemented utilizing the object oriented programming (OOP). At startup the oe_serv process instantiates an OEServ object, an OEConfig object, and a Logger object. The OEserv object is an instance of a main server class for starting the server processing including reading in the configuration and setting up the listening sockets. The OEConfig object is used to read specific StarOE configuration values, and the Logger object logs informational, warning, and/or error messages to a file. The OEConfig and Logger objects are typically set to Global so that all classes may access configuration values and log messages. The OEServ object then creates a Rogue Wave RWServerSocket object which creates a server socket connection and begins listening for client connections. The Rogue Wave Net++ library is typically used for socket communications.

The OEServ object includes a RWServerSocket object which waits to accept a connection from a client. When a client connects, a ClientHdlr object may be instantiated to handle the client connection. The ClientHdlr inherits from the Child class which when constructed, forks a new process for handling the client connection.

The ClientHdlr object reads the message header and determines how much to read for the following message request which includes message request header and body. The message request is read and the message id is interrogated. A switch statement on the message id determines how the message may be processed. The proper classes may then be instantiated to handle the request and determine the response. When the response is determined by the proper class, the ClientHdlr then may create a message header and response header including a return code and/or an error code. The ClientHdlr then sends the message back to the client with the response. The ClientHdlr typically need not know the contents of the request or the response message, instead, the ClientHdlr may act as a conduit for passing messages from the client to a proper class to handle the request, and relay the response from the proper class back to the client.

Some examples of the proper classes include the TollfreeSecurity class, the Application class, and the Hierarchy class. In addition, the server includes many other classes for processing specific functions. The TollfreeSecurity class encapsulates Tollfree security and access into the database tables having tollfree security detail information. When the ClientHdlr receives a Get User Tollfree Security request from the client, the TollfreeSecurity class is instantiated. The Get() method in this class access the database tables and returns the list of corp ids and tollfree numbers for this user. The ClientHdlr formats the data into a reponse message and returns the message back to the client.

The Application class encapsulates the interface into the database table having applications information. Any reading or writing to this table may be handled by the methods within this class. When the ClientHdlr receives a Get Application Request, the Application class is typically instantiated. The Get() method in this class accesses the Applications table in the database and return the list of application codes and their descriptions. The ClientHdlr then formats the data into a reponse message and returns the message back to the client.

The Hierarchy class encapsulates all hierarchy information and the methods to access this

data in the database. When the ClientHdlr receives a Get Hierarchy List request, the Hierarchy class is instantiated. The Get() method in this class determines which Hierarchy product is to be retrieved (e.g., Tollfree, Vnet/CVNS, or Vision) and returns the appropriate information. The ClientHdlr then formats the data into a response message and returns the message back to the client. The details of the message format, including request and response messages, are described with reference to Figure 13.

Order entry process via StarOE

Figure 10 is an illustrative example of an order entry process flow in the system of the present invention involving the functionalities described above. This process flow is used whenever a customer needs to interact with the StarOE system, whether it be as a new customer, or existing customer adding application services, or an existing customer changing individual passwords or work assignments for a specific application. The order entry process begins when a customer places an order for addition, modification or deletion of services, as shown in step 220. In step 222, an accounting team uses a new Web-based interface to enter the customer information. Typically, on a nightly basis, one or more e-mails may be generated to OEHub for Internet MCI access as shown in step 224, and to networkMCI Interact OEHubs as shown in step 232. At step 226, if the order is for a new customer, or if the existing customer needs a new browser package, then the OEHub handling Internet MCI service enters information from the e-mail to add or delete the customer. StarOE server then passes on mailing information to the fulfillment house, as indicated at step 228, which mails a package including the browser to the new customer to complete the process. The customer typically receives the browser package in 7-10 days. The fulfillment processing will be described in more detail with reference to Figure 14.

In step 234, the OEHubs enter data received from the e-mail. In step 236, if a new customer was added, the OEHubs add the customer information into the mainframe system via a direct interface as shown in steps 238 and 240. In steps 242 and 244, the new customer and service application information is added into the StarOE server database.

In step 246, additional steps are performed by StarOE if necessary for each application. For example, if an added service is a Toll Free Network Manager (TFNM), additional flags are tagged and stored in the application's server as shown in steps 248 to

258. In step 260, when the process of adding a new customer is complete, a fulfillment process, another component of the StarOE sends a welcome letter to the customer's inbox created as part of the addition process and transmits the new customer information to a fulfillment house as shown in steps 262 and 264. In step 266, the OEHub notifies account team that the order is complete.

In response to the transmission from the fulfillment process, the fulfillment house, in step 268, produces and mails a welcome letter that includes the user ID, a password to be used for the first logon and a list of applications the customer ordered. In step 270, if for any reason, the fulfillment process did not complete, a fulfillment representative contacts customer services, in step 272. In step 274, the customer services resolves issues by manually performing the fulfillment functions.

In step 276, if new services were added by an existing customer, the services added are updated to the mainframe system. In step 278, the StarOE database is also updated with the additional services. In step 280, if a service application requires specific added processing, for example, the TFNM, the additional processing are performed in step 248 through 266, as described previously.

Figure 11 is a high level input process flow diagram, illustrating inputs to the system of the present invention. The present invention handles a wide variety of key function for the suite of network applications. Each application will, herein forth, be also referred to as a fulfilling system, having a fulfilling client and a fulfilling server. The system of the present invention handles security and authentication requests from both the client and server sides of each fulfilling system as shown in 282a-d and 284. These requests are automatically generated whenever the customer makes a request of the server. For example, they are generated when a customer clicks on the icon for a service such as TFNM.

In addition, when a customer first logs on, the customer is presented with a dialog box prompting for user ID and password. When the customer clicks a submit button, for example, the backplane (or platform) verifies the customer is valid by inquiring with the StarOE system as shown in 286. The return response is either "invalid user/password" or "valid user". When the customer has been authenticated, the customer is then presented with a list of authorized applications. This list determines which buttons, for example, representing each application are active,

thus controlling customer access to products and services.

In addition, also shown in 286, the customer is issued a temporary password with the customer's fulfillment package. This temporary password is used to log into the system the first time. During the initial logon process, the customer is prompted to create a new password, which replaces the temporary password in the database. Moreover, the customer may be required to change the password every 30 days, for example, for security reasons.

Information may also be entered and requested by a number of sites other than a user platform. For example, OE Hubs 288 may enter information directly into the StarOE database to register new customers to the integrated suite of network applications. They may also access the data in StarOE directly to modify customer information, and to add or remove subscribed services.

Other inputs to the StarOE server may include entitlement data from the individual fulfilling systems 290a, 290b. For example, the mainframe 290a may send the appropriate hierarchy of tollfree numbers for a specific customer in response to registry message registering the new customer to the mainframe 290a. The hierarchy of tollfree numbers describes the new customer's entitlements to the TFNM services. This hierarchy may be used by other services in the integrated suite of network applications, for example, a Reporting application.

As described above, the StarOE is a repository for customer authentication and entitlement information. These authentication and entitlement data are usually transmitted from mainframe systems 292, 292a, 292b. For example, the mainframe system 292 generates two sets of hierarchy files on a daily basis. One set comprises deltas only, the other comprises a full hierarchy. Notification is made to the StarOE when these are available. At a set time each day, for example, 6:00 A.M., the StarOE, the reconciliation process more specifically, picks up the files via FTP and either replaces the previous day's hierarchy file with the new information (full file) or updates the previous day's file using the files having the updated data only. During this process, the reconciliation process first locks the database so that no other requests for data services may be made.

Figure 12 is an output process flow diagram, illustrating outputs from the system of the present invention. The StarOE outputs various responses to the requesting systems and processes. An example of an output is an authentication response to the client side of the individual applications 292a-c as well as

the backplane platform 294. In addition, a list of accessible applications for a given customer, is output to the backplane platform 294. The StarOE also outputs various updated data to database systems associated with specific individual applications 296a-c in the suite of network applications. In addition, the individual fulfilling systems 302a-d receive messages from the StarOE regarding modifications effected by a customer interaction. For example, as part of the reconciliation process, the StarOE may pass a list of toll free numbers to be deleted from Traffic View. These numbers represent the services which are discontinued. Upon receipt of this information, the Traffic View server sends another message to a system responsible for collecting call detail information which system then discontinues collection of call data for the numbers deleted. Another example output to individual fulfilling system is hierarchy data to Reporting fulfilling system 203c when a customer requests reports. The customer hierarchy data is sent in real time by the StarOE for up-to-date report information.

Additionally, the fulfillment house receives transactions from the StarOE. The StarOE, for example, by a fulfillment process, pushes a file including new customers to the fulfillment house 298, for instance, via FTP. This file includes customer mailing information, user name and password. The fulfillment house 298 creates a welcome package including the customer's user name and temporary password, and sends the package. The customer receives this package and uses the temporary password to logon to the system of integrated suite of network applications the first time. During the initial logon process, the user may be prompted to create a new password, which replaces the temporary password in the StarOE database. Upon logon, the user may find another welcome letter which lists the subscribed applications. This welcome letter was generated by StarOE server via e-mail when the order entry process was completed, and went to the Inbox 300 at that time.

StarOE transaction messages

Figure 13 illustrates message formats used in transaction requests and responses communicated to and from the client and server applications in the system of the present invention. As described with reference to Figure 6, the client applications typically send messages to the StarOE via the dispatcher 206 Figure 6. The messages are generally a stream of bytes including characters and each message typically has an order and format. The messages may also include fields of primitive types such as int or short.

The client applications, typically implemented in languages such as Java, encapsulate the message structures into class object. The application server programs including the StarOE, typically implemented using low level compiler languages such as C or C++, receives the message objects from the client applications as a stream of bytes and map the message objects into appropriate C structures. Typically, all the structures are aligned on byte boundary, byte aligned and in network byte order, so that Java objects may be mapped correctly into the C structures on the StarOE.

The StarOE messages have fixed formats. For example, for a request transaction, the request message 308 is preceded by a message header 304a and a request header 306. A response message 312 is preceded by a message header 304b and a response header 310.

The message header 304a, 304b is sent as the first part of all messages sent to or from the StarOE so that the receiver may know the size of the message being sent. This mechanism allows the server or client to read the proper number of bytes. When the OE server receives a message, a communications layer of the StarOE strips off the message header structure and adds it to all responses associated with this message.

The request header 306 follows the message header 304a on all messages sent to the StarOE. The StarOE examines the request header 306 for determining the type and version of message. It allows the StarOE to map the message to the correct C structure for subsequent processing. Similarly, response header 310 follows the message header 304b on all messages sent from the StarOE. The response header 310 precedes all response messages.

The StarOE client program typically implements message transactions using class objects. Figure 9 illustrates messages class diagram utilized in the present invention. The messaging classes 570

hide the complexities involved in issuing a transaction to the StarOE. Common functions are implemented in the base message classes encapsulating much of the effort required to build message classes. Message classes are created generally for requests and responses distinctly.

The request and response classes derive from the common base class OEMsg 572. The class OEMsg 572 encapsulates the common message header required by the StarOE. The classes OERqstMsg 574 and OERspMsg 576 serve as common base classes for the request and response messages respectively. All request messages also implement the COMessage interface. COMessage functions provide the necessary hooks to send and receive messages from the client platform to the back-end application servers. All StarOE messages use the synchronous transaction mechanism, wherein the transactions block until a response is received.

The following codes illustrates retrieval of user information using the order entry message classes.

```
OEUsrInfoRqst usrInfo = new
     OEUsrInfoRqst("abecar", session);
if (usrInfo.doIt() == true) {
     OEUsrInfoRsp uinfo =
                              (OEUsrInfoRsp)usrIn
                              fo.getResponseObj()
                              ;
     OEUsrInfo info = uinfo.getUsrInfoObj();
     System.out.println("First name:"
                              +info.firstName);
}
else
     System.out.println("Failed to retrieve
user information");
```

Once a request object is built, the method doIt() on the OERqstMsg 574 class must be invoked to initiate the transaction. The method doIt() may block until a response is received. A boolean return code is returned by doIt(). To obtain the response data or detailed error code information the method getResponseObj() must be invoked to return the built response object.

Specific transaction messages communicated in the system of the present invention include "validate user message" for authenticating the customer's username/password pair, "get user application list message" for retrieving a list of applications to which the customer is entitled, and "update password message" for changing the customer's password. These transaction messages are typically communicated between the StarOE and the backplane unit. These messages may also be used by other

application services to verify entitlements in the event an extra level of security is desired.

Examples of transaction messages communicated between the StarOE server and the StarOE client include "get user info" for retrieving customer's information such as name, address, timezone for the location of the customer. Another example of a transaction message communicated to the StarOE system by all the application clients and servers may be a "get user security message" for retrieving customer's security information including the userid, usertype which describes whether the customer is a regular user or an administrator, and access id for describing the customer's access level. The system of the present invention includes a number of such transaction messages described above for specific type of requests and responses communicated to and from the OE system and, moreover, additional message specific transactions may be added as necessary.

StarOE interface to fulfillment house

Figure 14 is a diagram illustrating data flow interface with the fulfillment house in the system of the present invention. Generally, customers are added by Order Entry, via StarOE client program 316. A fulfillment process typically sends information on new customers to the fulfillment house 298 on a nightly basis and also sends a news message comprising welcome letters to each new customer via the Inbox 300. If a new application is added to an existing customer then only a message is sent to the customer via the Inbox 300.

The fulfillment kit comprises a welcome letter and an inbox message. The fulfillment process in the StarOE server 202 typically runs on a daily basis. The information needed for the fulfillment process is generally stored in a database 314, Informix for example, in table forms. Such tables may include a user, configuration, and enterprise user application tables. The information or data is retrieved, configured and computed from the tables and put into a temporary fulfillment table which is used for the collection of the fulfillment information. From the fulfillment table the process creates the inbox letters and records for the fulfillment file. The enterprise user application table usually includes the status of whether the requested applications have been fulfilled. The fulfillment file is transferred, using file transfer protocol (FTP), for example, to the fulfillment group 298 for creation of the welcome letters.

Figure 8 illustrates an example of a welcome letter 80. The welcome letter 80, which is in a language that the customer requested, comprises the customer's id and password, URLs pointing to a language specific Web page where a customer may typically start up the system of the present invention which the customer has ordered.

StarOE interface to StarWRS

Figure 15 is a diagram illustrating data flow interface with the Reporting application. The Reporting application is one of the applications available as part of the integrated suite of network applications. Figure 15 shows StarOE server's interface to various systems in order to provide the Reporting application the appropriate information for producing reports. The reconciliation process generally loads data, such as the toll free hierarchies and pick list information in the StarOE database from the mainframe systems 320. The data are required to setup the security profile for the Reporting application. The loading may be performed periodically, for example, on a nightly basis.

The StarOE client 316 may modify a customer's Reporting Application security profile and add, update, or delete the application and hierarchy access. The hierarchy and security are stored in StarOE database 314. When a customer modifies the customer's Reporting Application security profile, a select set of data are sent to appropriate servers for real time processing. For example, if the modified security profile includes Un-priced Reports, then the records in the security profile with these reports are sent in realtime to the Traffic View Server 322, such that the Traffic View Server 322 may begin collecting data as soon as the customer makes the modification.

A Reporting application's client program, report requester 324a as well as the servers 324b, 324c also retrieves a customer's profile for reporting purposes. For example, a report scheduler 324b, a part of the reporting servers, verifies via a direct connection with the StarOE database 314, the customer's access level to the hierarchy before scheduling report requests. The Inbox server 300a also requests paging and e-mail information from StarOE server 202 for optional report notifications.

StarOE interface to TFNM

Figure 16 is a diagram illustrating the data flow interface with the Toll Free Network Manager application. The Toll Free Network Manager is one of the applications available as part of the integrated suite of network applications. Figure 16 shows StarOE server's interface to various systems in order to provide the Toll Free Network Manager application the appropriate information toll free servicing. The data hierarchies necessary for the Toll Free Network Manager are retrieved by the reconciliation process via FTP nightly, for example, from the mainframe system 320 and stored in the StarOE database 314. The data hierarchies retrieved are used to setup the security profile for the Toll Free Network Manager application 326a, 326b. Information setup in the security profile are sent to the mainframe systems 320 by the StarOE server 202. For example, the corp ids that are setup at a global level in StarOE database 314 for use by the Toll Free Network Manager application represent participants of the Toll Free Network Manager and are flagged as such in the mainframe system when the information is received. Therefore, upon receiving the ids the mainframe system 320, for any reason, may not delete these corp ids while they are flagged. Additionally, the Toll Free Network Manager server 326a sends requests to the StarOE server 314 to get a Toll Free Network Manager customer's security. This is done by formulating a get user security message in a transaction request as described in reference to Figure 13.

StarOE interface to Broadband

Figure 17 is a diagram illustrating data flow interface with the Broadband application. The Broadband is one of the applications available as part of the integrated suite of network applications. Figure 17 shows the StarOE server's interface to various systems in order to provide the Broadband application the appropriate information for broadband servicing. Customer's user and security profile for the Broadband application is added by Order Entry 330. When a Broadband customer's profile is added, changed, or deleted, the Broadband server 328a needs to be notified of the operation so that the server 328a can either begin or stop collecting data for the specific customer's bill ids. Typically the notification is done by a message in the form of a file. The file is typically placed in a known directory and is transferred to the Broadband server 328a via FTP process. The file transferred generally includes all the current Broadband customers with their bill ids.

While the invention has been particularly shown and described with respect to preferred embodiments thereof, it will be understood by those skilled in the art that the foregoing and other changes in form and details may be made therein without departing from the spirit and scope of the invention. More specifically, the system administrative and order entry system described may apply to any generic applications available via the Web and is not limited to the telecommunications applications explained above for exemplary purposes.

CLAIMS

What is claimed is:

1. A Web-based centralized authentication and entitlement administration system for enabling a customer to enter orders over the Internet from a client terminal for one or more application services available at an enterprise, the system comprising:

a client browser application located at the client terminal and providing an integrated interface to the one or more application services, the client browser application interacting with the customer;

an order entry server located at the enterprise, the order entry server for communicating over the Internet with the client browser application to provide authentication and entitlement information associated with the customer, wherein the client browser application obtains the authentication and entitlement information associated with the customer and applies the authentication information in validating the customer before enabling the customer to access the enterprise, the client browser application enabling the customer to access only those application services to which the customer is entitled; and

an order entry object initiated by the client browser application, for enabling presentation of entry options for the customer, the entry options including adding a new order entry, modifying an existing order entry, and canceling an order entry, the order entry object further communicating customer entry of a specific entry option to the order entry server,

whereby the customer is enabled via the integrated interface to enter new orders, modify existing orders, and cancel orders for the application services within the customer entitlements.

2. The system as claimed in claim 1, wherein the order entry server further comprises a customer profile database for storing the authentication and entitlement information.

3. The system as claimed in claim 1, wherein the order entry object further comprises a first device for communicating new order entries, modified entries, and canceled order entries in response to customer entry of a specific option, and wherein the order entry server further comprises a device for accepting and storing the customer entries.

4. The system as claimed in claim 3, wherein the first device for communicating comprises a plurality of messaging classes, the plurality of messaging classes including:

base message classes for encapsulating standards required for communicating between the client browser application and the order entry server;

request and response classes derived from the base message classes for handling a plurality of request and response transactions communicated between the client browser application and the order entry server, which request and response transactions include the new order entries, modified entries, and canceled order entries associated with the customer selection of a specific option.

5. The system as claimed in claim 1, wherein the order entry object includes an entry application downloaded from the enterprise, the entry application running in its own window frame.

6. The system as claimed in claim 2, wherein the order entry server further comprises a fulfillment process for retrieving from the customer profile information associated with newly added customers, and wherein the fulfillment process notifies and electronically transmits the information to a fulfillment house responsible for sending subscriptions packages to new customers.

7. The system as claimed in claim 6, wherein the fulfillment process runs periodically on a pre-defined time basis.

8. The system as claimed in claim 6, wherein the fulfillment process further sends the subscription packages with a welcome message to a message center created for and associated with the new customers.

9. The system as claimed in claim 2, wherein the order entry server further comprises a reconciliation process for updating and synchronizing data stored in the customer profile with data stored in mainframe systems.

10. The system as claimed in claim 9, wherein the reconciliation process runs periodically on a pre-defined time basis.

11. The system as claimed in claim 9, wherein the reconciliation process further updates and synchronizes the customer profile data with data

stored in servers associated with the application services.

12. The system as claimed in claim 11, wherein the reconciliation process updates and synchronizes data in real time, wherein the application services use real time data when processing requests by the customer.

13. The system as claimed in claim 2, wherein the order entry server further comprises a billing process for automatically directing customer bills to a billing stream specified in the customer profile.

14. The system as claimed in claim 2, wherein the order entry object presents the entry options in a tree format.

15. The system as claimed in claim 14, wherein a first tree level represents an enterprise, a second tree level represents users under the enterprise, and a third tree level represents application services accessible by the users.

16. The system as claimed in claim 15, wherein the second tree level further comprises a new user node for enabling addition of a new user under an enterprise.

17. The system as claimed in claim 14, wherein a first tree level represents an enterprise, a second tree level represents a corp, and a third tree level represents toll free numbers subscribed by the corp.

18. The system as claimed in claim 14, wherein the order entry object enables modification of entries associated with a node in the tree when the node selected.

19. The system as claimed in claim 18, wherein the modification of entries includes modifying reporting options for toll free numbers.

20. The system as claimed in claim 18, wherein the modification of entries includes modifying user access levels associated with the application services.

21. The system as claimed in claim 18, wherein the modification of entries includes modifying

target billing streams where customer bills are transmitted.

22. A method of providing Internet enabled centralized authentication and entitlement administration services for enabling a customer to enter orders over the Internet from a client workstation for one or more application services available from an enterprise, the method comprising:

presenting to a customer a client browser application having an integrated interface to the one or more application services;

communicating authentication requests and responses over the Internet to an order entry server located at the enterprise;

authenticating the customer at the client workstation with authentication information received from the order entry server;

enabling the customer to access predetermined application services according to an entitlement response received from the order entry server;

enabling presentation of entry options for the customer, the entry options including adding a new order entry, modifying an existing order entry, and canceling an order entry; and then

communicating customer selected entry options to the order entry server,

wherein the customer is enabled via the integrated interface to enter new orders, modify existing orders, and cancel orders for the application services within the customer entitlements.

23. The method according to claim 22, wherein the method further comprises:

storing authentication and entitlement information in a customer profile database at the enterprise.

24. The method according to claim 22, wherein the method further comprises updating and synchronizing the customer profile database with data in at least one mainframe system providing an application service to the customer.

25. The method according to claim 22, wherein the method further comprises transmitting data associated with a customer selected entry option to a server associated with a respective application service.

26. The method according to claim 22, wherein the method further comprises presenting entry entitlements to the customer in a tree format.

27. The method according to claim 23, wherein the method further comprises directing customer bills to billing streams specified in the customer profile database.

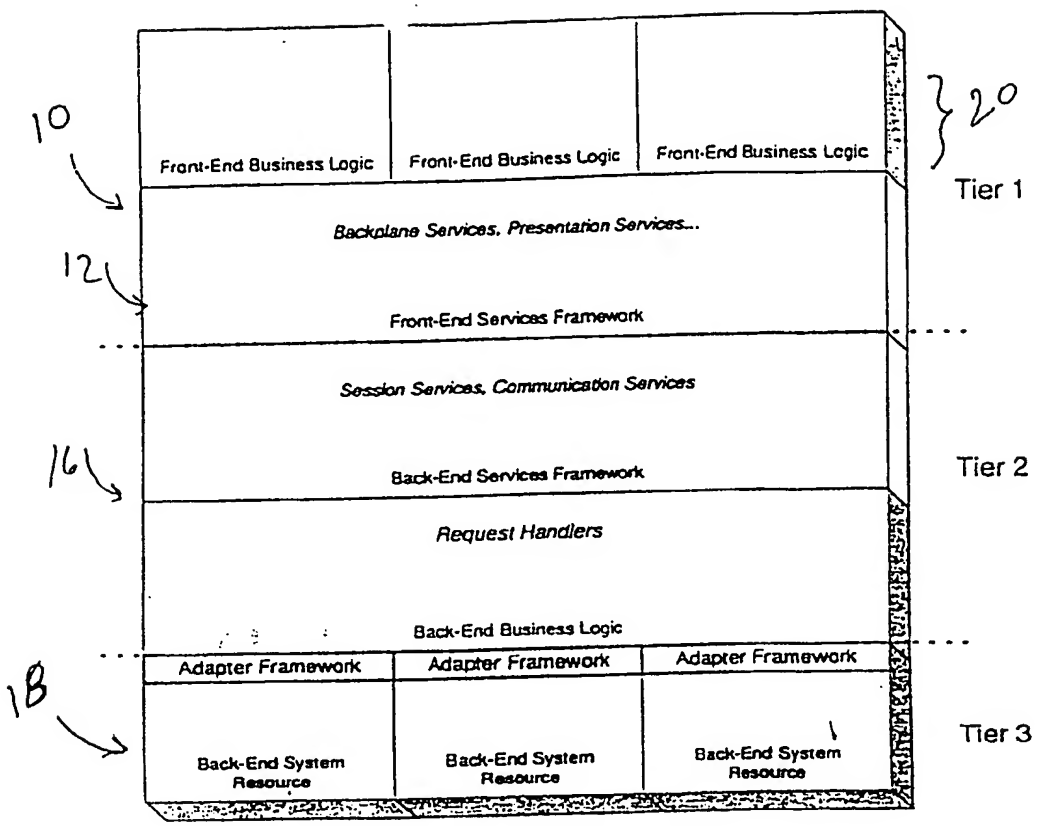

10
Front-End Business Logic
Front-End Business Logic
Front-End Business Logic
20
Tier 1
Backplane Services, Presentation Services...
12
Front-End Services Framework
Session Services, Communication Services
16
Back-End Services Framework
Tier 2
Request Handlers
Back-End Business Logic
Adapter Framework
Adapter Framework
Adapter Framework
18
Back-End System Resource
Back-End System Resource
Back-End System Resource
Tier 3

Fig 1

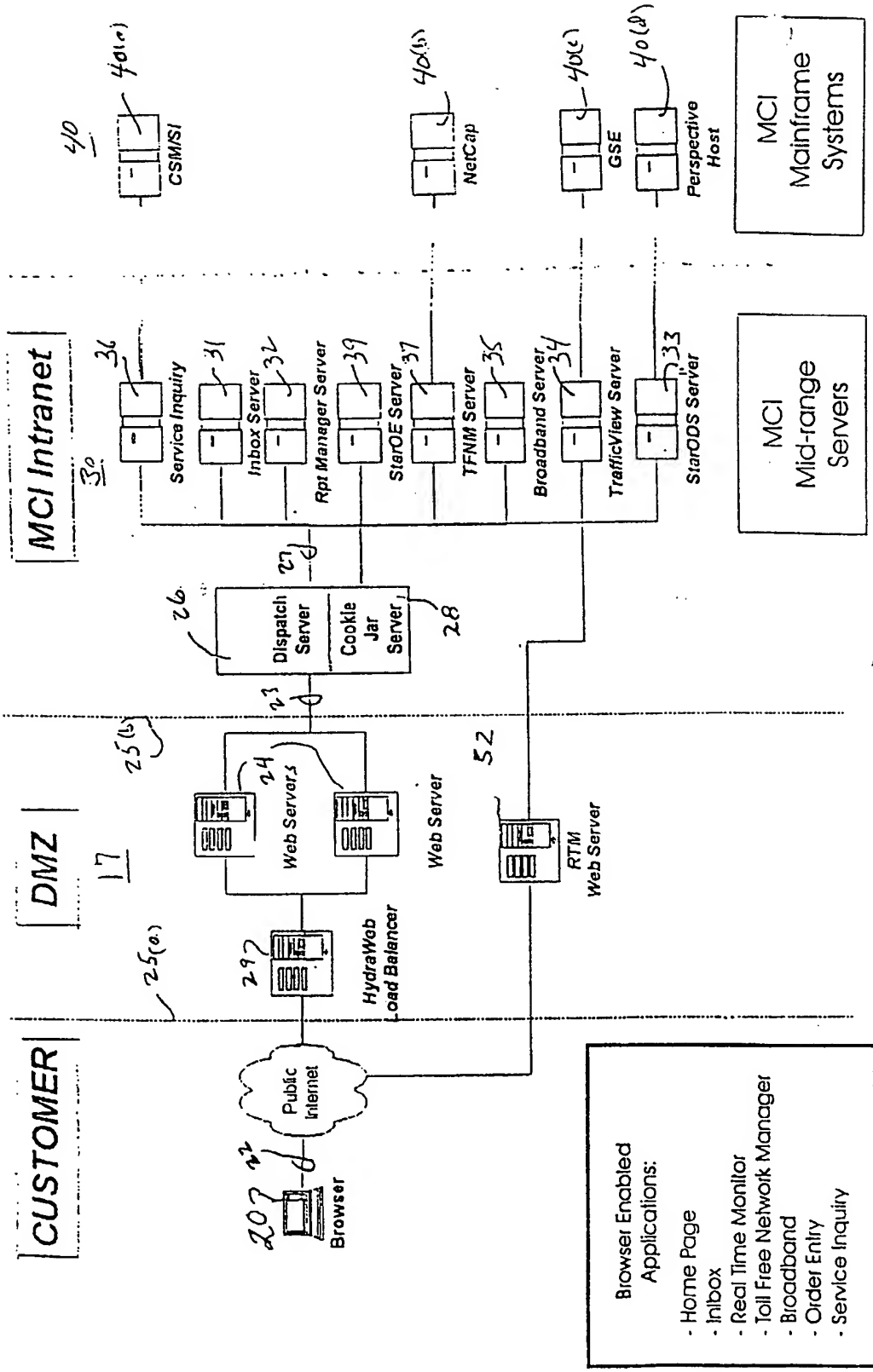

CUSTOMER
DMZ
MCI Intranet
MCI Mainframe Systems
MCI Mid-range Servers
Browser
Public Internet
HydraWeb Load Balancer
Web Servers
Web Server
RTM Web Server
Dispatch Server
Cookie Jar Server
Service Inquiry
Inbox Server
Rpt Manager Server
StarOE Server
TFNM Server
Broadband Server
TrafficView Server
StarODS Server
CSMIS
NetCap
GSE
Perspective Host
Browser Enabled Applications:
- Home Page
- Inlbox
- Real Time Monitor
- Toll Free Network Manager
- Broadband
- Order Entry
- Service Inquiry

Figure 2

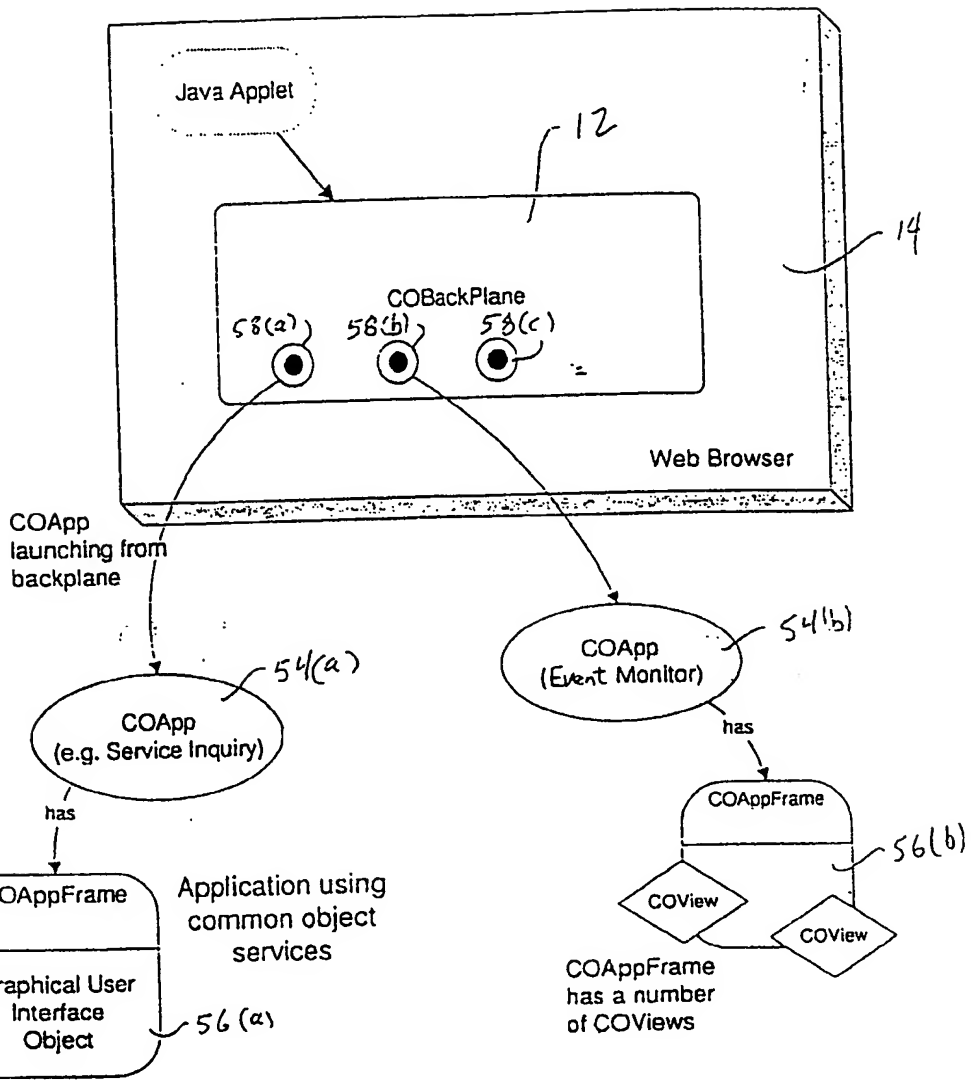

Java Applet
12
14
COBackPlane
58(a)
58(b)
58(c)
Web Browser
COApp launching from backplane
54(a)
COApp (e.g. Service Inquiry)
has
COAppFrame
Graphical User Interface Object
56(a)
Application using common object services
54(b)
COApp (Event Monitor)
has
COAppFrame
56(b)
COView
COView
COAppFrame has a number of COViews

Figure 3

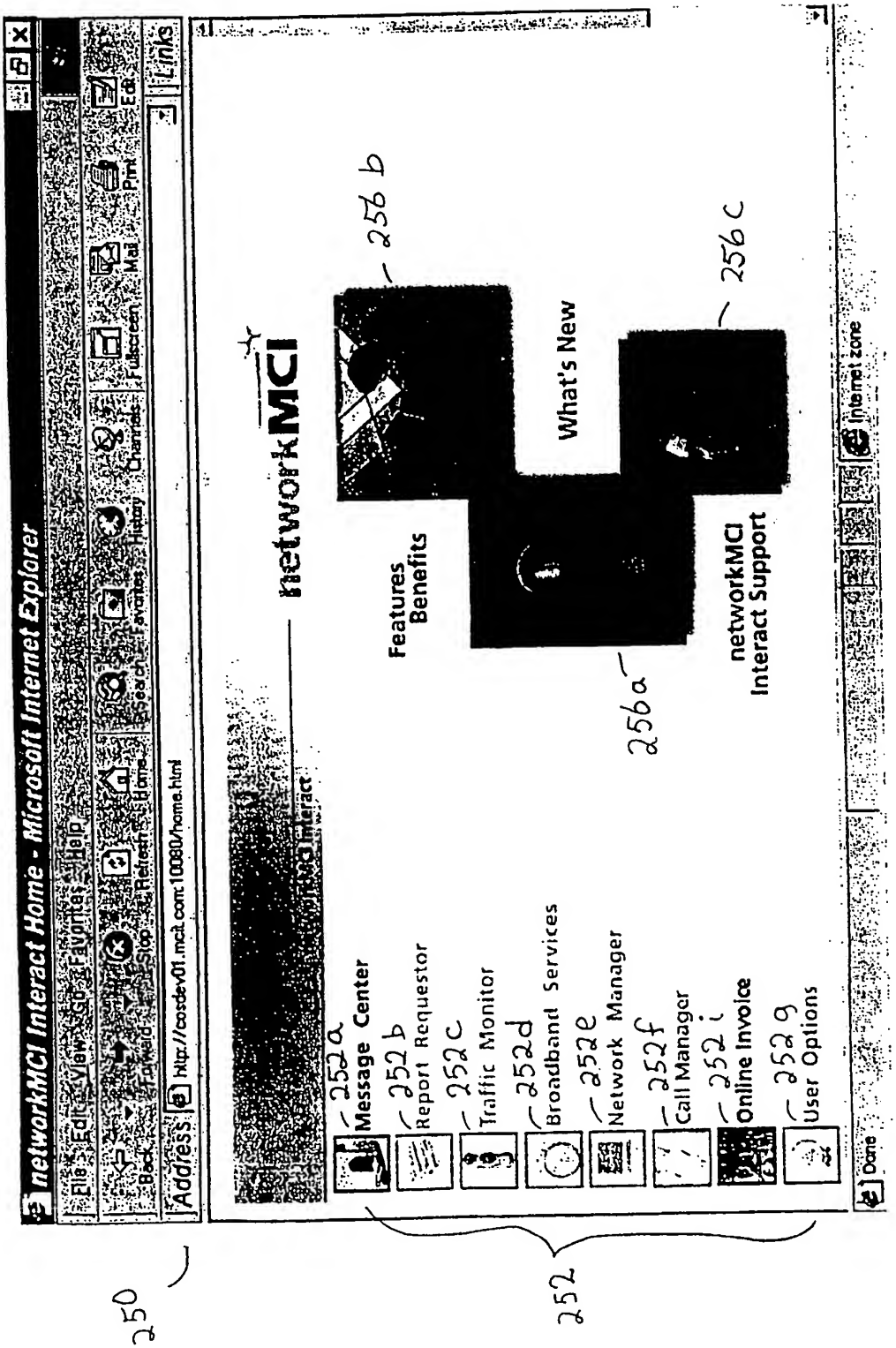

networkMCI Interact Home - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen Mail Print Edit
Address http://casdev01.mci.com:10080/home.html
Links
networkMCI
Interact
Message Center
Report Requestor
Traffic Monitor
Broadband Services
Network Manager
Call Manager
Online Invoice
User Options
Features Benefits
What's New
networkMCI Interact Support
Done
Internet zone
250
252
252a
252b
252c
252d
252e
252f
252i
252g
256a
256b
256c

Figure 4

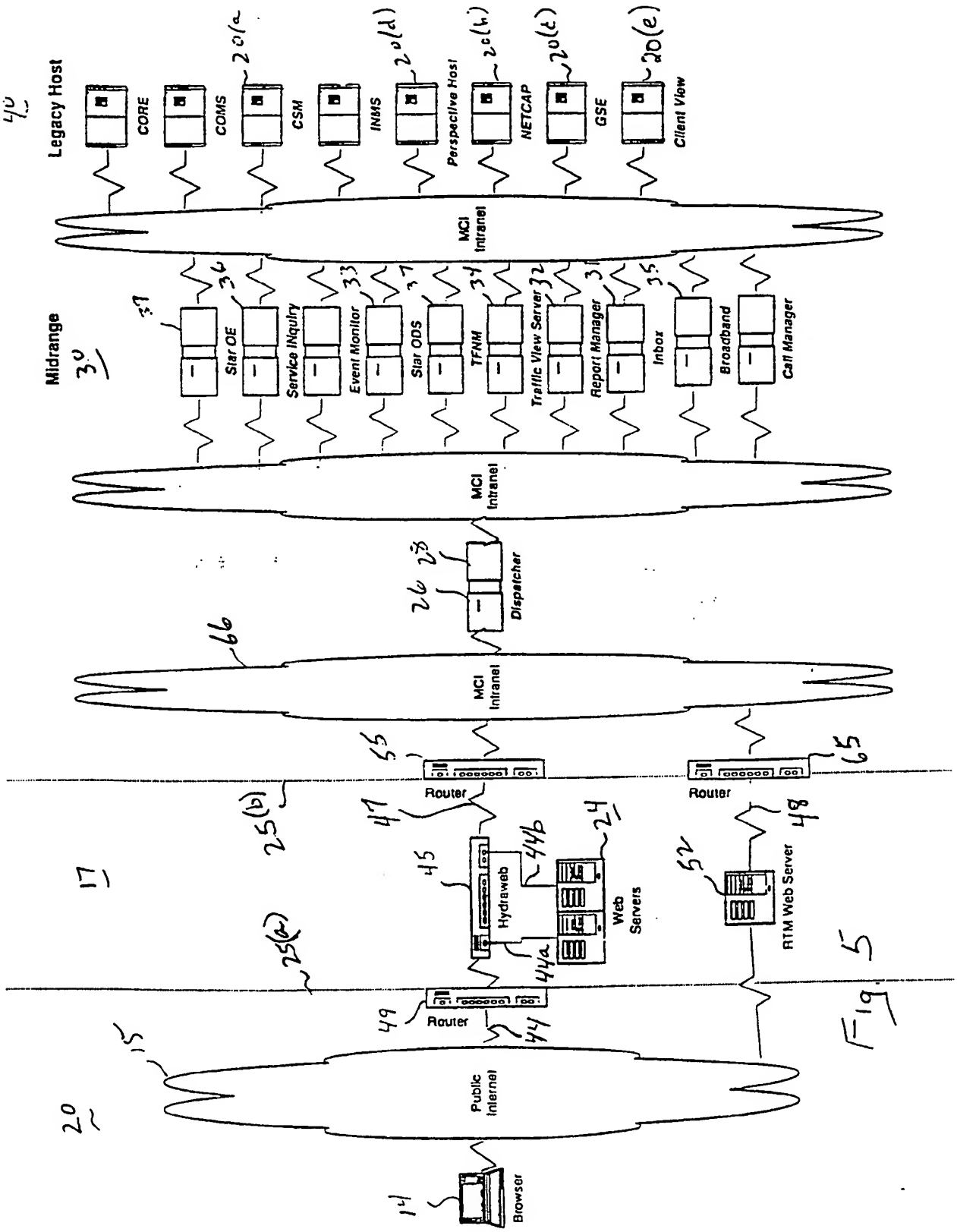

Legacy Host
CORE
COMS
CSM
INMS
Perspective Host
NETCAP
GSE
Client View
MCI Intranet
Midrange
Star OE
Service INquiry
Event Monitor
Star ODS
TFNM
Traffic View Server
Report Manager
Inbox
Broadband
Call Manager
MCI Intranet
Dispatcher
MCI Intranet
Router
Router
Hydraweb
Web Servers
RTM Web Server
Router
Public Internet
Browser

Fig 5

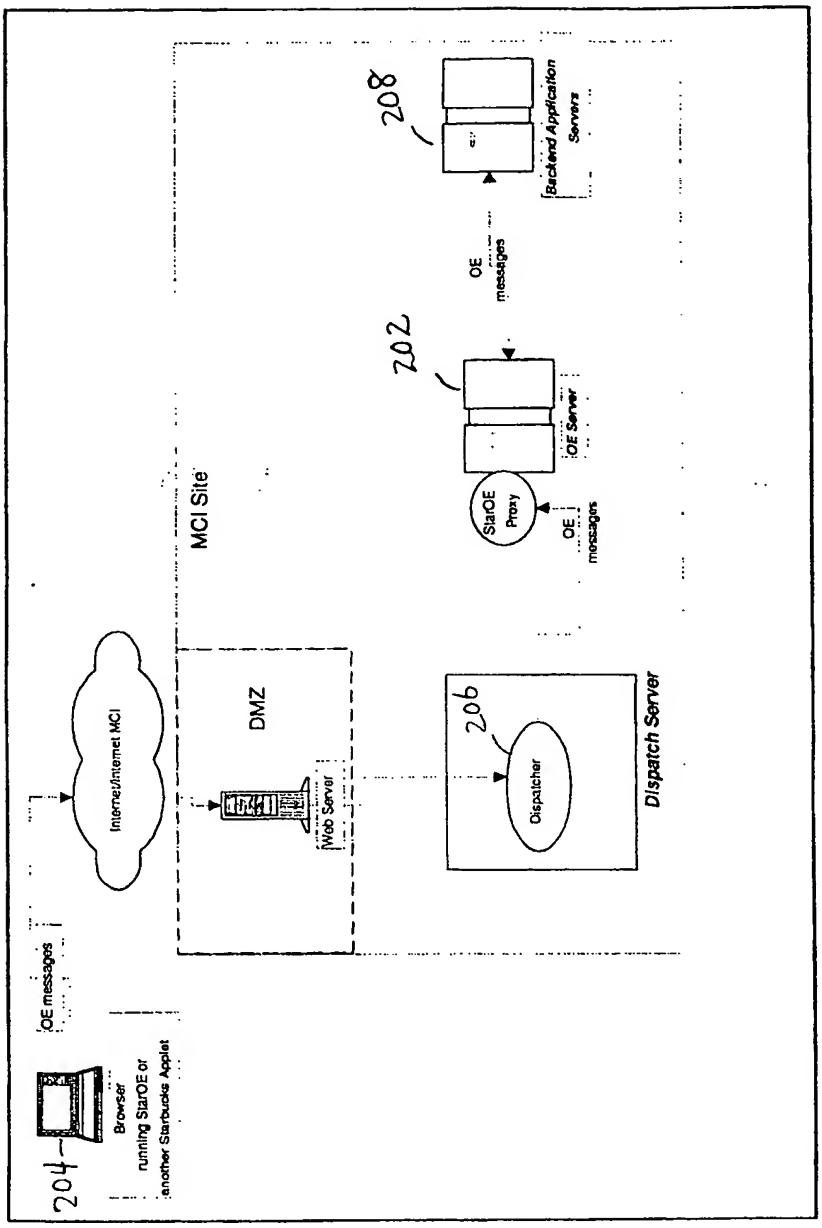

208
Backend Application Servers
OE messages
202
OE Server
StarOE Proxy
OE messages
MCI Site
DMZ
Web Server
Internet/Internet MCI
206
Dispatcher
Dispatch Server
OE messages
204
Browser running StarOE or another Starbucks Applet

Figure 6

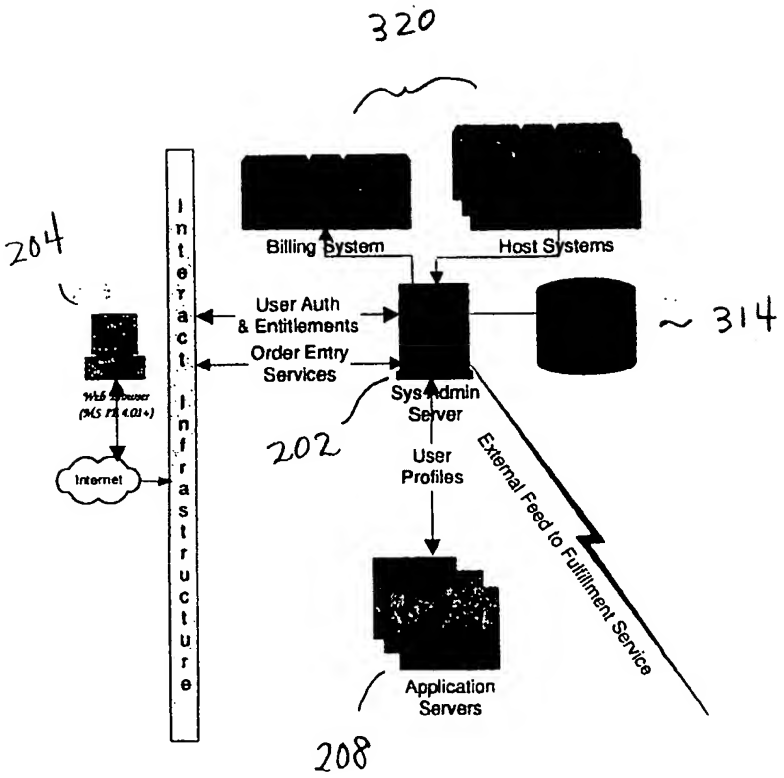

320
204
Billing System
Host Systems
User Auth
& Entitlements
Order Entry
Services
314
Sys Admin
Server
202
Internet
User
Profiles
External Feed to Fulfillment Service
Application
Servers
208
Interact Infrastructure

Figure 7

networkMCI

MCI
Customer Applications Support
2424 Garden of the Gods Road
A3-718
Colorado Springs, CO 80919

June 1, 1998

Bryan Fenley
MCI
2424 Garden of the God
colorado Springs, CO 80919

Enterprise ID: 10379399

Dear networkMCI Interact^{SM} Subscriber,

Welcome to networkMCI Interact. Now you have everything you need to monitor and manage your company's MCI services via the World Wide Web.*

To access networkMCI Interact's integrated suite of applications, you must have Internet access. For your convenience, we are sending you Internet access software under separate cover in case you currently do not have an Internet Service provider.

Because you are a networkMCI Interact Subscriber, we are offering you FREE unlimited local Internet access when you register for networkMCI Internet^{SM}—a savings of $19.95 per month! Each month, this amount will appear on your invoices as a full credit. In addition 800/888 toll-free access is also available at the rate of $5.95 per hour. When you receive your networkMCI Internet CD-ROM package, simply refer to the Getting Started Guide for instructions on how to install and register for networkMCI Internet.**

Once you have access to the Internet, you must log-on to the networkMCI Interact Web site, https://www.interact.mci.com, using the personal User ID and Temporary Password listed below.

LOG-ON INFORMATION

User ID: tollfree
Temporary Password: password

When you first log-on to the networkMCI Interact home page, instructions will be provided for personalizing your password. Also, for more information pertaining to the applications you have ordered, simply go to the Message Center located on the networkMCI Interact home page.

If you have questions or need technical support for networkMCI Interact, please call 1-800-326-2775. For networkMCI Internet technical support, please call 1-800-326-2775. For a complete listing of applications available through networkMCI Interact, please contact your MCI account representative.

Thank you for choosing networkMCI Interact.

Sincerely,

Ronald J. McMurtrie
Vice President, Product, Local, and Global Marketing

* MCI's Internet browser utilizes Microsoft Internet Explorer which is a registered trademark of Microsoft Corporation in the United States and /or other countries.
**If you choose to use a browser other than networkMCI 4.01 or Microsoft Internet Explorer 4.01, we cannot guarantee the performance of, or provide technical support for that software.

80

Figure 8

31297.11.97

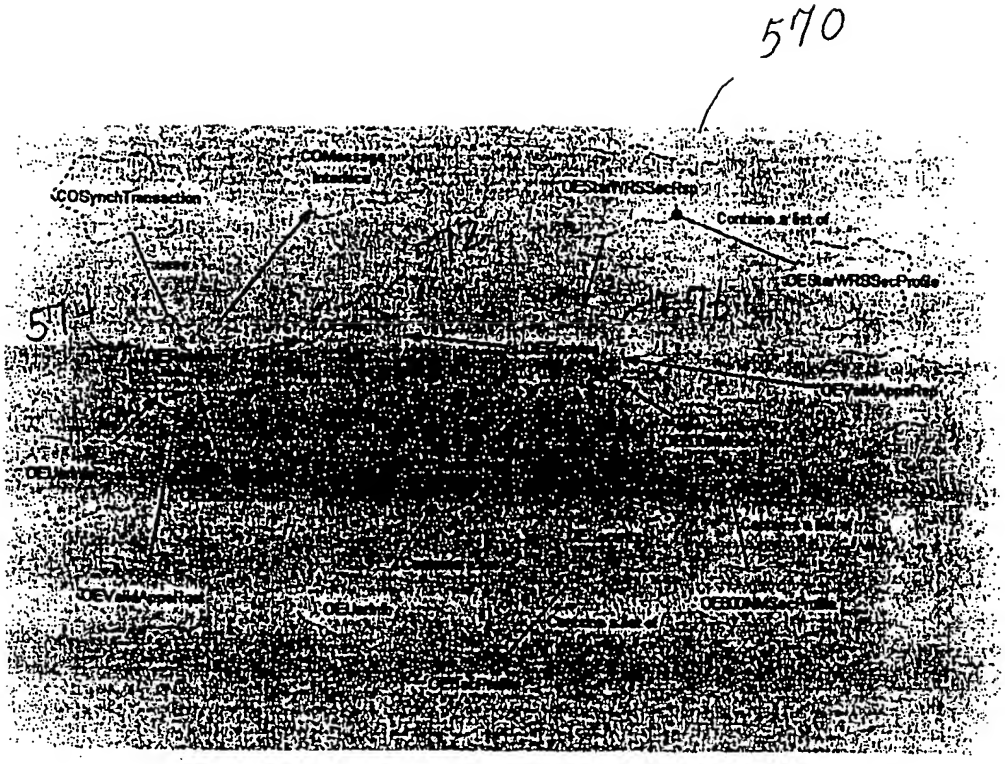


Figure 9

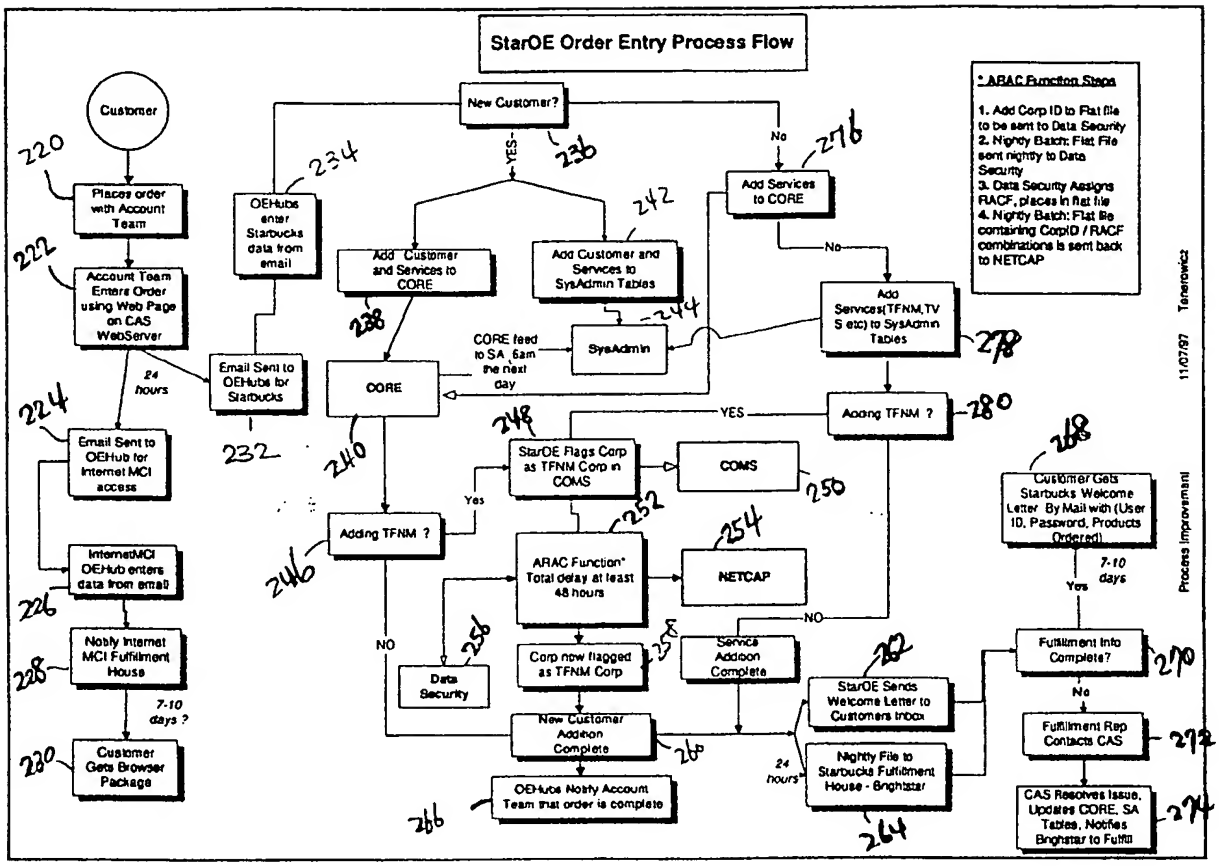

StarOE Order Entry Process Flow
Customer
New Customer?
YES
No
220
Places order with Account Team
222
Account Team Enters Order using Web Page on CAS WebServer
24 hours
224
Email Sent to OEHub for Internet MCI access
InternetMCI OEHub enters data from email
226
Notify Internet MCI Fulfillment House
228
7-10 days ?
230
Customer Gets Browser Package
OEHubs enter Starbucks data from email
234
Email Sent to OEHubs for Starbucks
232
Add Customer and Services to CORE
236
Add Customer and Services to SysAdmin Tables
242
238
CORE
CORE feed to SA 6am the next day
SysAdmin
244
240
Adding TFNM ?
246
Yes
NO
StarOE Flags Corp as TFNM Corp in COMS
248
COMS
250
ARAC Function* Total delay at least 48 hours
252
NETCAP
254
Data Security
256
Corp now flagged as TFNM Corp
258
New Customer Addson Complete
260
OEHubs Notify Account Team that order is complete
266
Add Services to CORE
276
No
Add Services(TFNM,TV S etc) to SysAdmin Tables
278
Adding TFNM ?
280
YES
NO
Service Addson Complete
282
StarOE Sends Welcome Letter to Customers Inbox
Nightly File to Starbucks Fulfillment House - Brightstar
24 hours
264
Customer Gets Starbucks Welcome Letter By Mail with (User ID, Password, Products Ordered)
268
7-10 days
Yes
Fulfillment Info Complete?
270
No
Fulfillment Rep Contacts CAS
272
CAS Resolves Issue, Updates CORE, SA Tables, Notifies Brightstar to Fulfill
274
* ARAC Function Steps
1. Add Corp ID to Flat file to be sent to Data Security
2. Nightly Batch: Flat File sent nightly to Data Security
3. Data Security Assigns RACF, places in flat file
4. Nightly Batch: Flat file containing CorpID / RACF combinations is sent back to NETCAP
11/07/97 Tenerowicz
Process Improvement

Figure 10

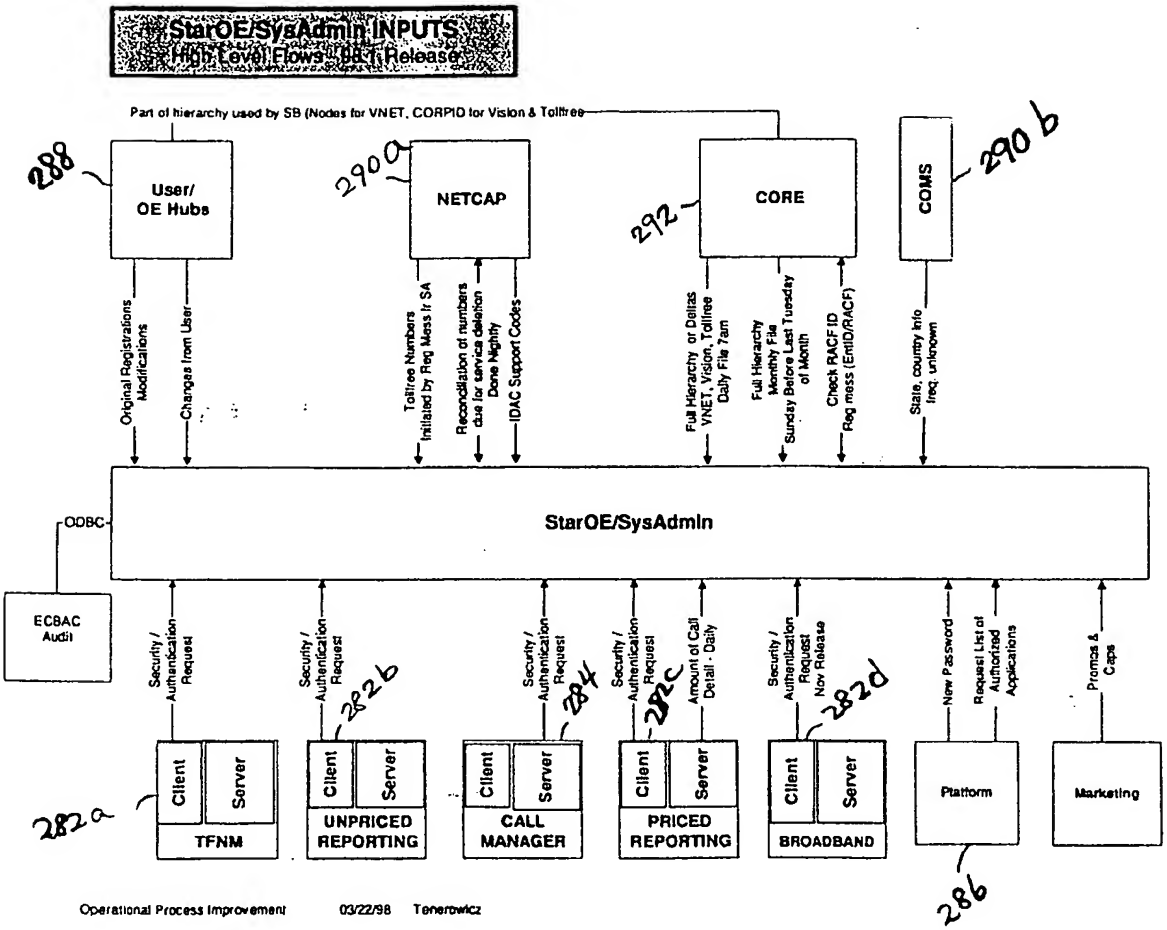

StarOE/SysAdmin INPUTS
High Level Flows - 98.1 Release
Part of hierarchy used by SB (Nodes for VNET, CORPID for Vision & Tollfree
288
User/
OE Hubs
290a
NETCAP
292
CORE
COMS
290b
Original Registrations
Modifications
Changes from User
Tollfree Numbers
Initialed by Req Mass Ir SA
Reconciliation of numbers
due for service deletion
Done Nightly
IDAC Support Codes
Full Hierarchy or Delizs
VNET, Vision, Tollfree
Daily File 7am
Full Hierarchy
Monthly File
Sunday Before Last Tuesday
of Month
Check RACF ID
Req mass (EmID/RACF)
State, country info
Ireq. unknown
ODBC
StarOE/SysAdmin
ECBAC
Audit
Security /
Authentication
Request
Amount of Call
Detail - Daily
New Password
Request List of
Authorized
Applications
Promos &
Caps
282a
282b
284
282c
282d
Client
Server
TFNM
UNPRICED
REPORTING
CALL
MANAGER
PRICED
REPORTING
BROADBAND
Platform
Marketing
286
Operational Process Improvement
03/22/98
Tenerowicz

Figure 11

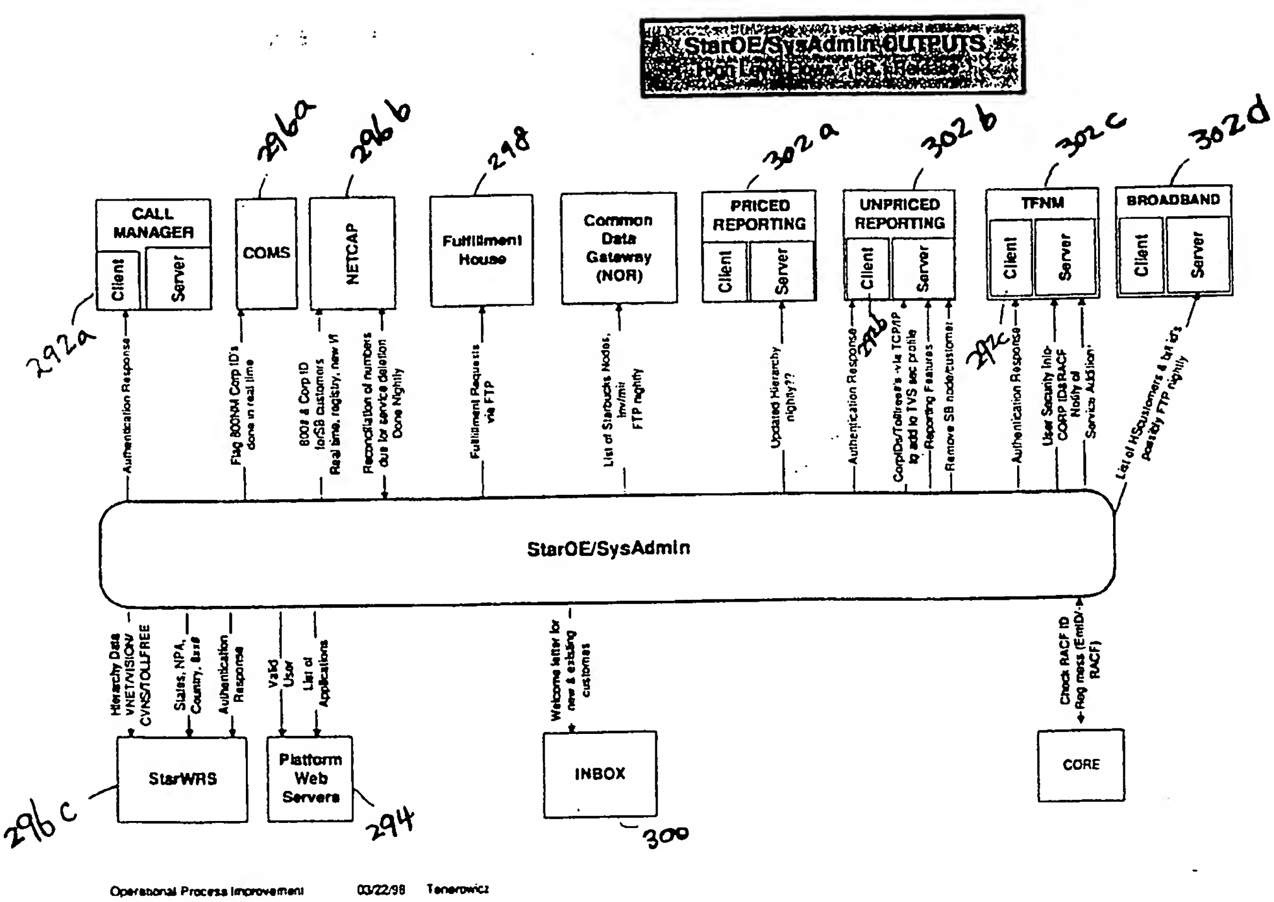

StarOE/SysAdmin OUTPUTS
CALL MANAGER
Client
Server
COMS
NETCAP
Fulfillment House
Common Data Gateway (NOR)
PRICED REPORTING
Client
Server
UNPRICED REPORTING
Client
Server
TFNM
Client
Server
BROADBAND
Client
Server
Authentication Response
Fulfillment Requests via FTP
Updated Hierarchy nightly??
StarOE/SysAdmin
StarWRS
Platform Web Servers
INBOX
CORE
Valid User
List of Applications
Authentication Response
292a
296a
296b
298
302a
302b
302c
302d
296c
294
300
Operational Process Improvement 03/22/98 Tenerowicz

Figure 12

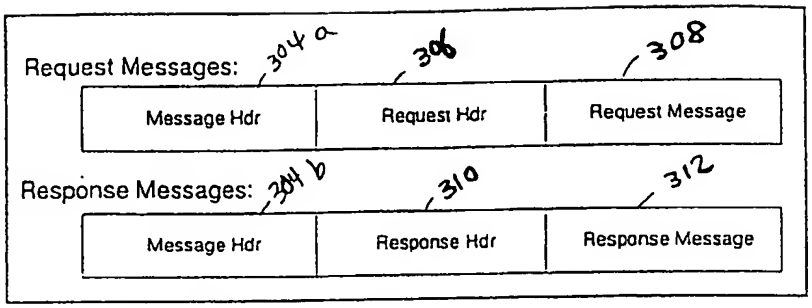

Request Messages:
304a
306
308
Message Hdr
Request Hdr
Request Message
Response Messages:
304b
310
312
Message Hdr
Response Hdr
Response Message

Figure 13

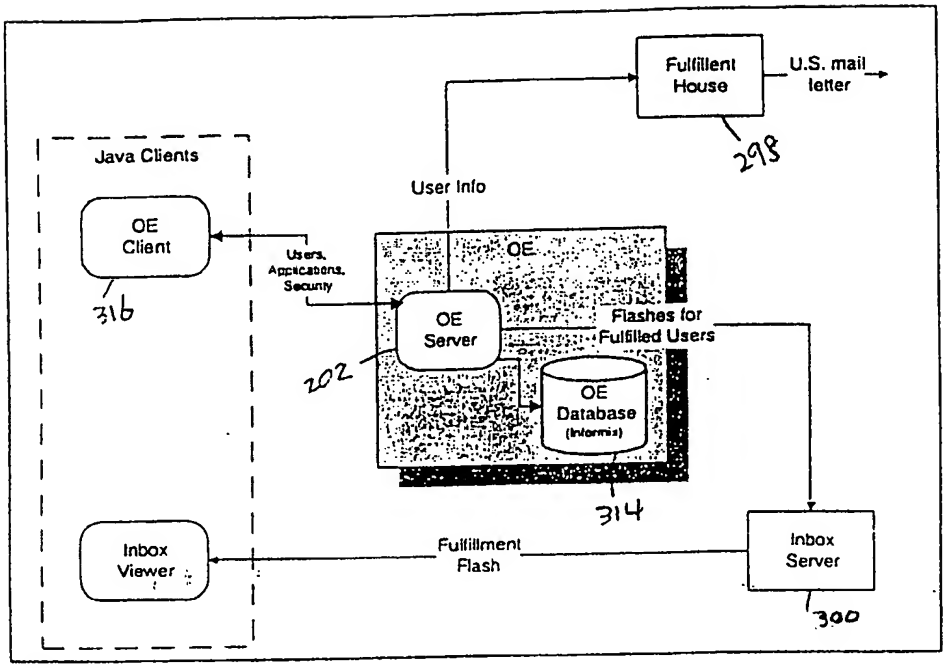

Fulfillent House
U.S. mail letter
298
Java Clients
User Info
OE Client
316
Users, Applications, Security
OE
OE Server
202
Flashes for Fulfilled Users
OE Database
(Informix)
314
Inbox Viewer
Fulfillment Flash
Inbox Server
300

Figure 14

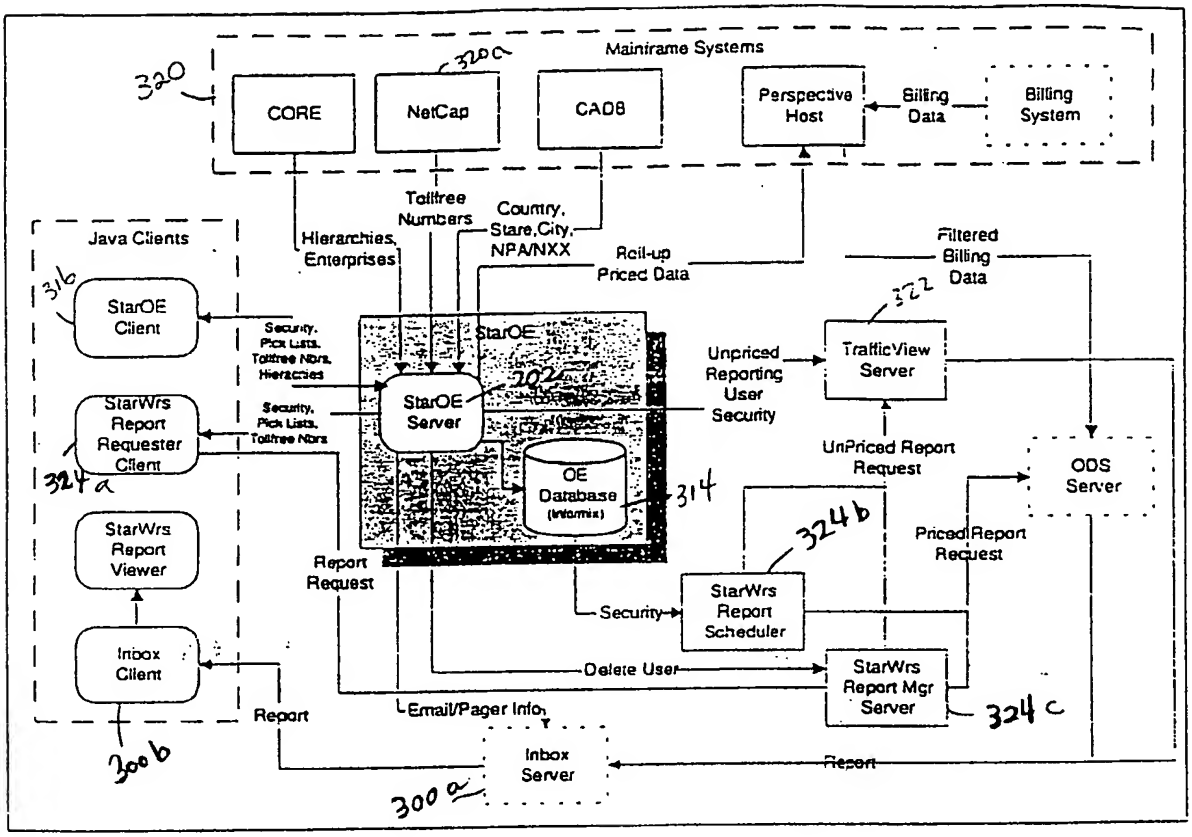

320
Mainframe Systems
320a
CORE
NetCap
CADB
Perspective Host
Billing Data
Billing System
Java Clients
316
StarOE Client
StarWrs Report Requester Client
324a
StarWrs Report Viewer
Inbox Client
300b
Tollfree Numbers
Country, State,City, NPA/NXX
Hierarchies, Enterprises
Roll-up Priced Data
Filtered Billing Data
322
Security, Pick Lists, Tollfree Nbrs, Hierarchies
Security, Pick Lists, Tollfree Nbrs
StarOE
StarOE Server
302
OE Database (Informix)
314
Unpriced Reporting User Security
TrafficView Server
UnPriced Report Request
ODS Server
324b
Priced Report Request
Report Request
Security
StarWrs Report Scheduler
Delete User
StarWrs Report Mgr Server
324c
Email/Pager Info
Report
Inbox Server
300a
Report

Figure 15

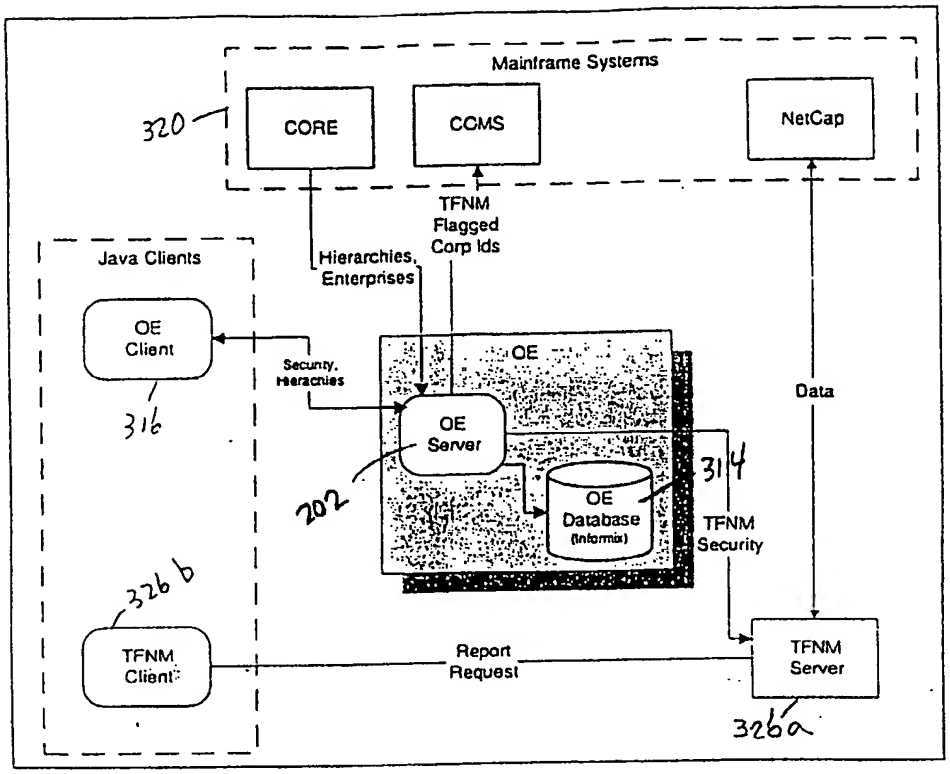

Mainframe Systems
320
CORE
CCMS
NetCap
TFNM
Flagged
Corp Ids
Hierarchies,
Enterprises
Java Clients
OE
Client
316
Security,
Hierarchies
OE
OE
Server
202
OE
Database
(Informix)
314
Data
TFNM
Security
326b
TFNM
Client
Report
Request
TFNM
Server
326a

Figure 16

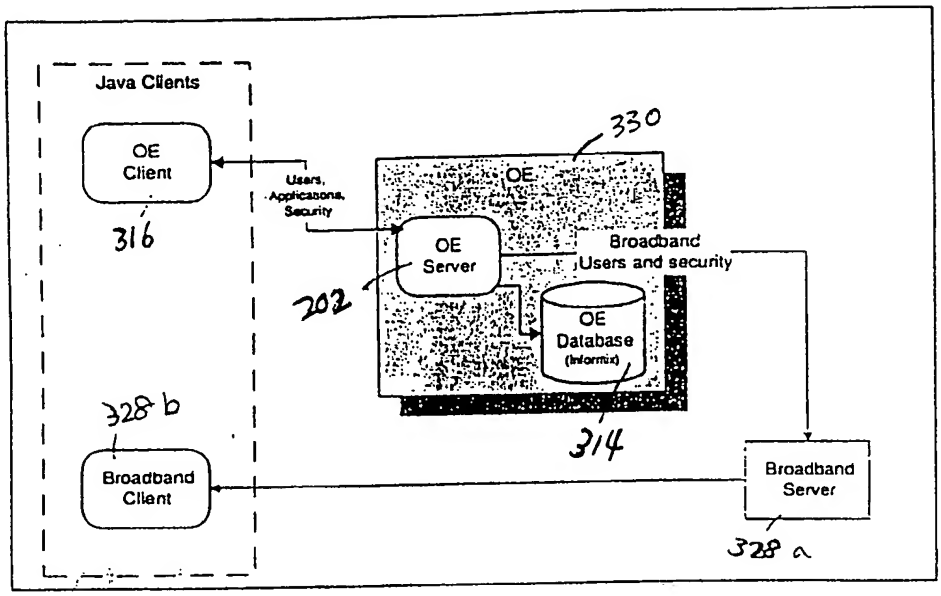

Java Clients
OE Client
316
Users, Applicasons, Security
330
OE
OE Server
202
Broadband Users and security
OE Database (Informix)
314
328 b
Broadband Client
Broadband Server
328 a

Figure 17

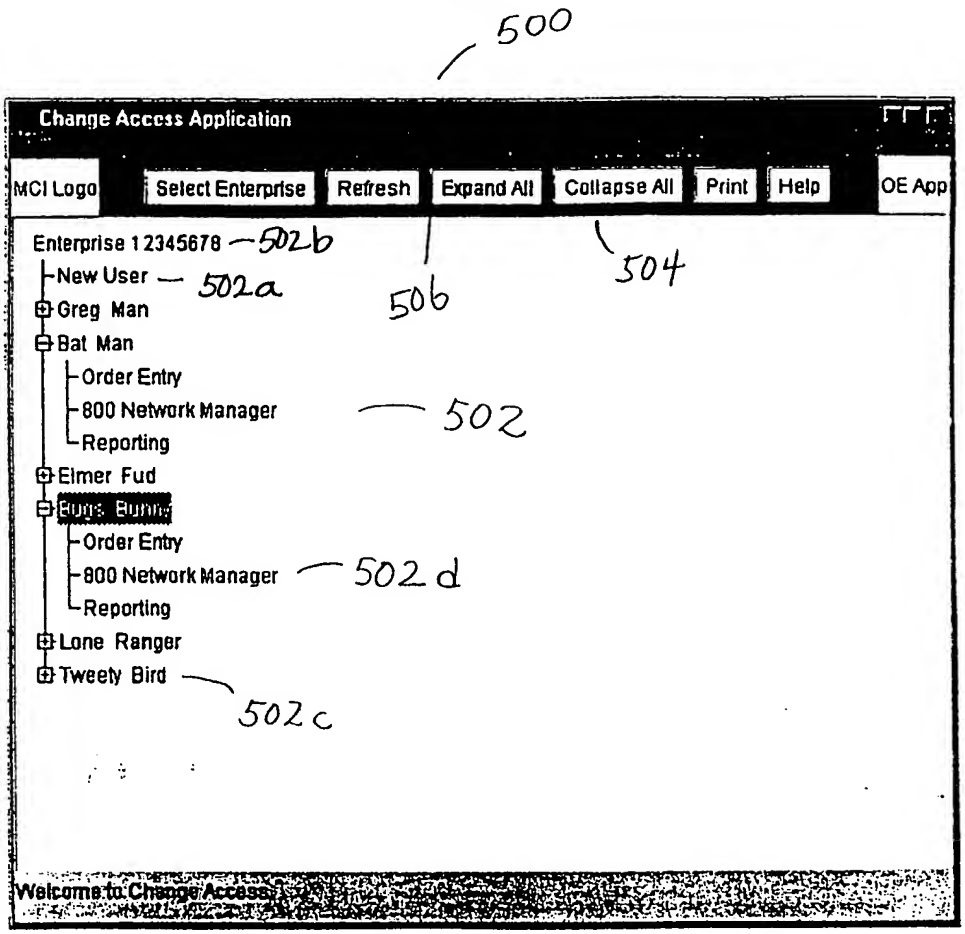

500
Change Access Application
MCI Logo
Select Enterprise
Refresh
Expand All
Collapse All
Print
Help
OE App
Enterprise 12345678
502b
New User
502a
Greg Man
Bat Man
Order Entry
800 Network Manager
Reporting
502
506
504
Elmer Fud
Bugs Bunny
Order Entry
800 Network Manager
Reporting
502d
Lone Ranger
Tweety Bird
502c
Welcome to Change Access

Figure 18

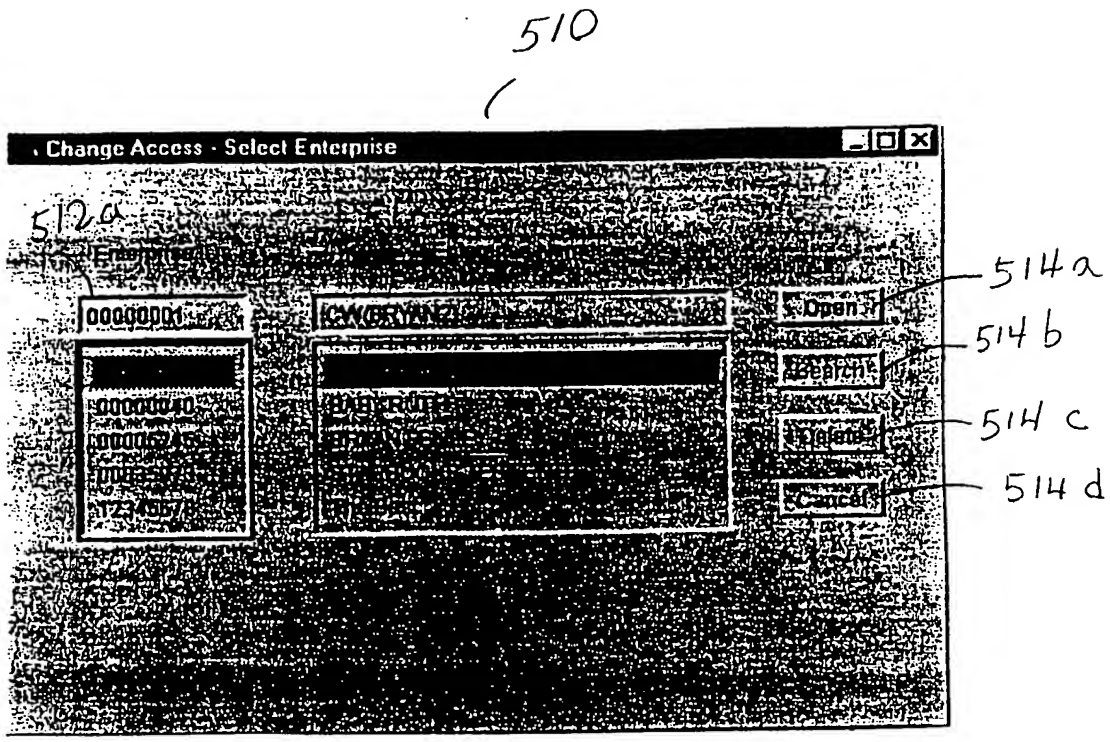

510
Change Access - Select Enterprise
512a
00000001
Open
Search
Options
Cancel
514a
514b
514c
514d

Figure 19

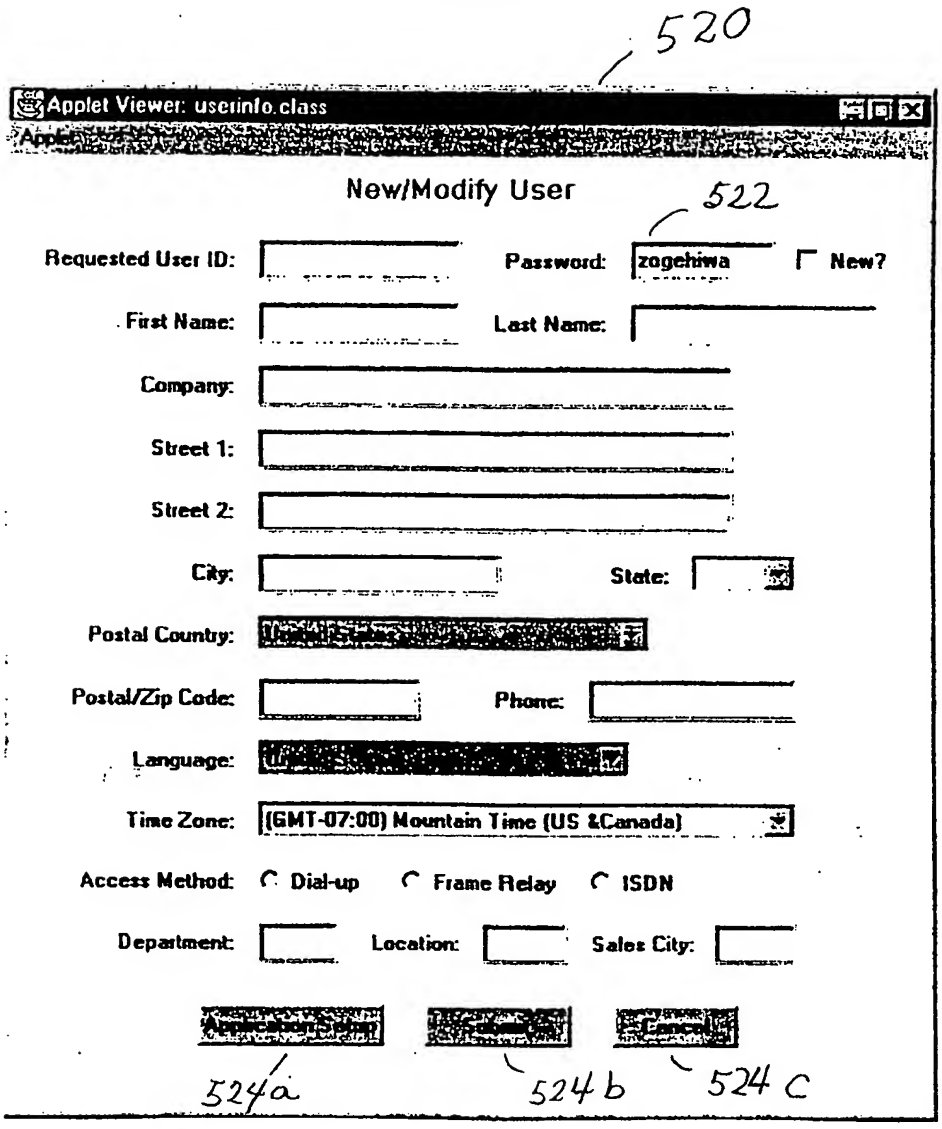

520
Applet Viewer: userinfo.class
New/Modify User
522
Requested User ID:
Password:
zogehiwa
New?
First Name:
Last Name:
Company:
Street 1:
Street 2:
City:
State:
Postal Country:
Postal/Zip Code:
Phone:
Language:
Time Zone:
(GMT-07:00) Mountain Time (US &Canada)
Access Method:
Dial-up
Frame Relay
ISDN
Department:
Location:
Sales City:
Cancel
524a
524b
524c

Figure 20

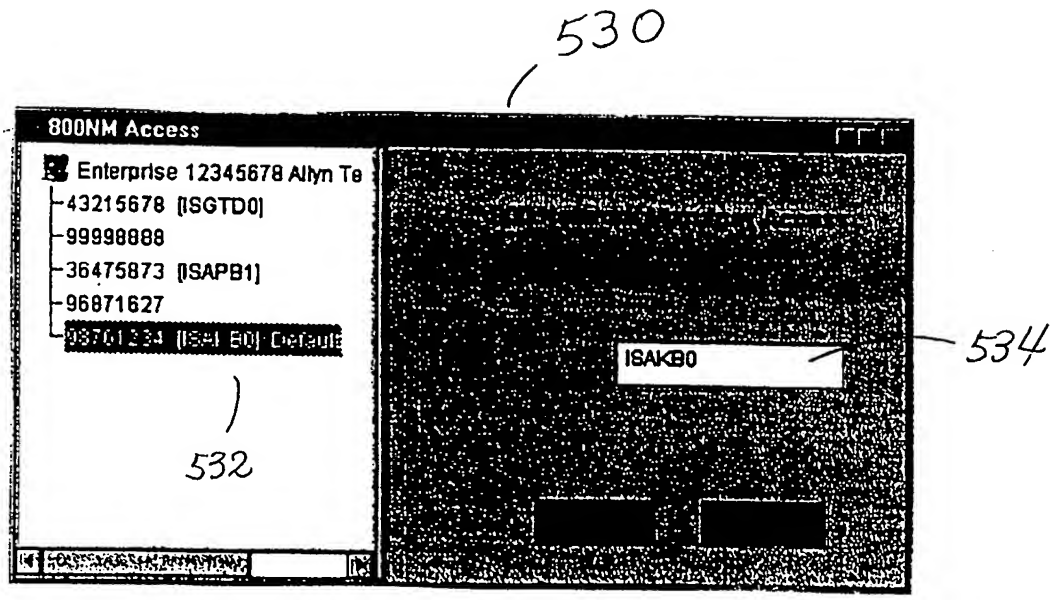

530
800NM Access
Enterprise 12345678 Allyn Te
43215678 [ISGTD0]
99998888
36475873 [ISAPB1]
96871627
ISAKB0
534
532

Figure 21

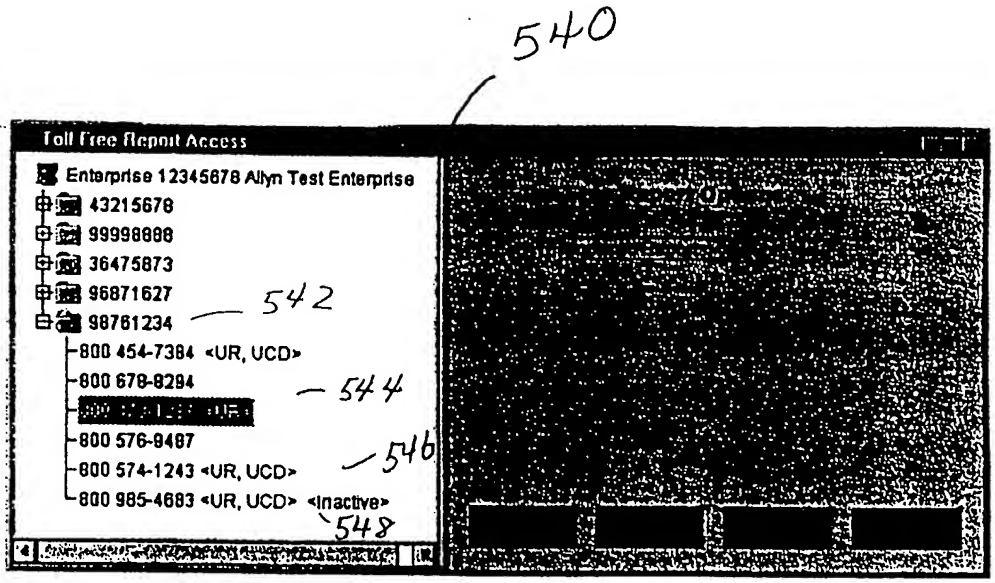

540
Toll Free Report Access
Enterprise 12345678 Allyn Test Enterprise
43215678
99998888
36475873
96871627
98761234
542
800 454-7384 <UR, UCD>
800 678-8294
544
800 576-9487
546
800 574-1243 <UR, UCD>
800 985-4683 <UR, UCD> <Inactive>
548

Figure 22

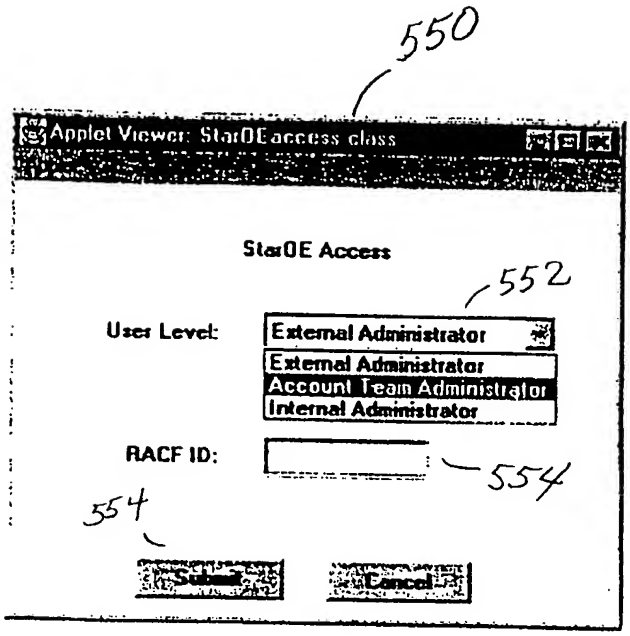

550
Applet Viewer: StarOEaccess class
StarOE Access
552
User Level:
External Administrator
External Administrator
Account Team Administrator
Internal Administrator
RACF ID:
554
554
Submit
Cancel

Figure 23

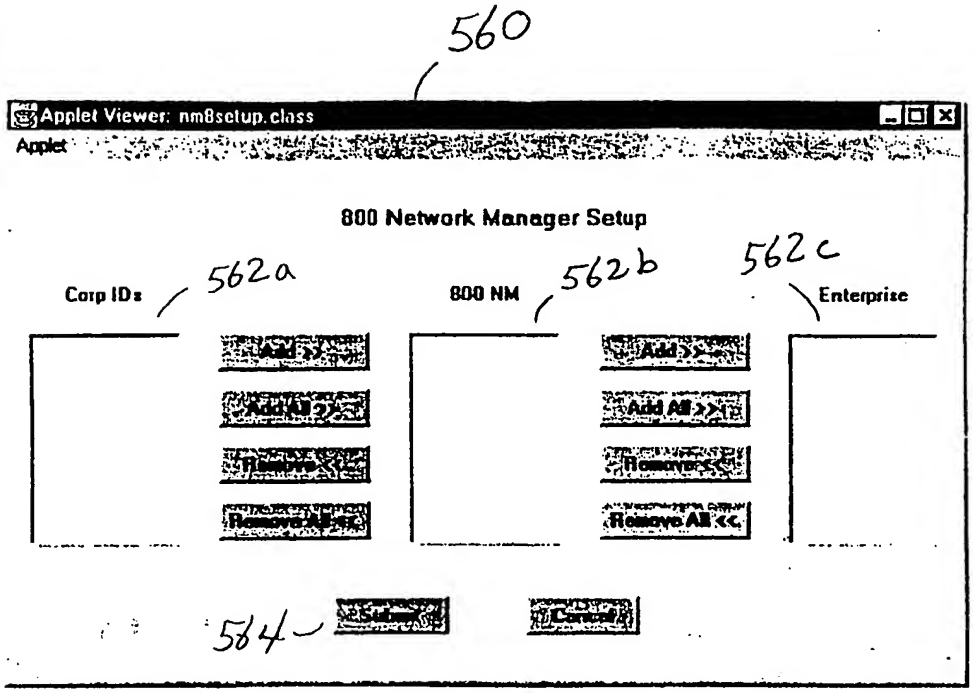

560
Applet Viewer: nm8setup.class
Applet
800 Network Manager Setup
Corp IDs
562a
800 NM
562b
562c
Enterprise
Add >>
Add All >>
Remove <<
Remove All <<
584
Submit
Cancel

Figure 24

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/20159

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 17/00
US CL : 395/200.54, 200.55, 186, 187.01; 380/25
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/200.54, 200.55, 186, 187.01; 380/25

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y,P	US 5,805,803 A (BIRRELL et al.) 08 September 1998, (08.09.98) cols. 1-6.	1-27
Y,P	US 5,826,029 A (GORE, JR. et al.) 20 October 1998, (20.10.98) cols. 2-4.	1-27
Y,P	US 5,793,964 A (ROGERS et al.) 11 August 1998, (11.08.98) Cols. 1-19.	1-27
Y,P	US 5,815,665 A (TEPER et al.) 29 September 1998, (29.08.98) cols.1-16.	1-27
Y	TANENBAUM COMPUTER NETWORKS PRENTICE HALL (New York) 1996 Pages 410-412	1-27

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"B"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 20 DECEMBER 1998	Date of mailing of the international search report 04MAR 1999
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer Ellis B. Ramirez Joni Hill Telephone No. (703) 305-9784

Form PCT/ISA/210 (second sheet)(July 1992)★